International Journal of Minerals, Metallurgy and Materials
Volume 26, Number 12, December 2019, Page 1485
https://doi.org/10.1007/s12613-019-1849-y

Dimensioning a stockpile operation using principal component analysis

Siyi Li, Marco de Werk, Louis St-Pierre, and Mustafa Kumral

Department of Mining and Materials Engineering, McGill University, 3450 University Street, Montreal, Quebec H3A 0E8, Canada

(Received: 31 December 2018; revised: 9 March 2019; accepted: 11 March 2019)

Abstract: Mineral processing plants generally have narrow tolerances for the grades of their input raw materials, so stockpiles are often maintained to reduce material variance and ensure consistency. However, designing stockpiles has often proven difficult when the input material consists of multiple sub-materials that have different levels of variances in their grades. In this paper, we address this issue by applying principal component analysis (PCA) to reduce the dimensions of the input data. The study was conducted in three steps. First, we applied PCA to the input data to transform them into a lower-dimension space while retaining 80% of the original variance. Next, we simulated a stockpile operation with various geometric stockpile configurations using a stockpile simulator in MATLAB. We used the variance reduction ratio as the primary criterion for evaluating the efficiency of the stockpiles. Finally, we used multiple regression to identify the relationships between stockpile efficiency and various design parameters and analyzed the regression results based on the original input variables and principal components. The results showed that PCA is indeed useful in solving a stockpile design problem that involves multiple correlated input-material grades.

Keywords: bed-blending; mining; stockpile; principal component analysis; multiple regression

1. Introduction

Bed-blending operations are applied in a variety of industries, including the mining industry, which uses stockpiles to homogenize and reduce the variability of raw materials before their delivery to mineral processing plants. The reason being that unfavorable residual variations inevitably persist even in materials from the same source, due to the discontinuous, cyclic, random, and autocorrelated nature of ore [1]. The optimization of processing efficiency relies heavily on homogenizing the input materials [2]. A bed-blending system has two phases. In the first phase, a stacker traverses the ground at a constant velocity along the stockpile, during which process materials are laid down on the same level as the stacker. As the stacker gradually reaches the end of the stockpile, it decelerates until it finally stops, before starting again to travel back in the opposite direction. In the reclaiming phase, a reclaimer (either a bucket-wheel or a harrow-type scraper) cuts slices of the stockpile perpendicular to the direction of stacking [3]. In the past, many researchers have worked towards optimizing the design of blending operations and have proposed various theories and methods [4–6]. In 1992, Gy introduced the idea of using the variance reduction ratio (VRR) to evaluate the effectiveness of blending [7]. Dowd [8] suggested the use of geostatistical approaches to improve stockpile design by predicting the output characteristics of given stockpile parameters. Kumral [9] incorporated multiple regression and genetic algorithms into the optimization of stockpile design. Recently, there has been an increasing number of statistical methods and mathematical models applied in the optimization of metallurgical and minerals engineering operations [10–13]. Designing a bed-blending operation would be relatively straightforward if just one mineral grade was of concern to the processing plant. However, this is rarely the case as raw material grades are multivariate in nature. For instance, certain types of iron ores can have more than six different chemical compositions that must be homogenized [14–15]. Therefore, challenges arise in situations where there are different levels of variations in the material grades that comprise the stockpile input.

In this research, we propose the utilization of principal component analysis (PCA), which is a dimension-reduction technique that is widely applied in many fields including image

Corresponding author: Mustafa Kumral E-mail: mustafa.kumral@mcgill.ca

and signal processing, statistical mechanics, and multivariate quality control. By introducing PCA to this problem, it is possible to greatly reduce the number of varying materials, while preserving most of the information from the original data, and thereby facilitate the design of stockpiles with minimal loss of information. We conducted this research in three main steps:

(1) We performed principal component analysis on the input data, projecting it to a lower-dimension space while retaining most of the original information. The input data used in this study is a serially correlated dataset with realistic statistical properties that could well occur in a real-world problem.

(2) We built a computer algorithm to simulate the process of bed-blending, which mimics stacking and reclaiming processes by laying down blocks of discrete unit weight and volume to form a cuboid and then slicing it across its length. The stockpile simulator computes the input and output variances of all the material grades, including those of the principal components.

(3) We used multiple regression to identify the relationships between the response and predictors, with the response being the VRR, and the predictors being the stockpile design parameters. We repeated this step for all the input materials including the principal components.

2. Methodology

2.1. Principle component analysis

PCA is a multivariate statistical tool that reduces p-dimensional correlated variables to a set of ordered and uncorrelated k-dimensional linear projections. Mathematically, this process is related to finding the spectral/Eigen decomposition of the positive-semidefinite variance–covariance matrix of the singular-value decomposition (SVD) of a rectangular data matrix.

2.1.1. Spectral decomposition of the variance–covariance matrix

Let $\boldsymbol{X} \in \mathbf{R}^{n \times p}$ be a data matrix where n represents the number of observations and p is the number of variables with $n > p$. $\boldsymbol{X}_\mathrm{c}$ is the mean-centered data matrix with $\boldsymbol{X}_\mathrm{c} = \boldsymbol{X} - 1_n \boldsymbol{\mu}^\mathrm{T}$, with 1_n being an $n \times 1$ column vector of 1 s and $\boldsymbol{\mu}^\mathrm{T}$ being the $1 \times p$ row vector denoting the variable means. Let $\boldsymbol{x}_\mathrm{c}$ be the row vector representing the variables $\boldsymbol{x}_\mathrm{c} = [x_1, x_2, \cdots, x_p]$. The variance–covariance matrix can then be found by the following:

$$\boldsymbol{S} = \frac{1}{n}\boldsymbol{X}_\mathrm{c}^\mathrm{T}\boldsymbol{X}_\mathrm{c} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_p^2 \end{bmatrix} \tag{1}$$

The variance–covariance matrix is symmetric and positive semidefinite with the diagonal entries being the variances of each variable and the rest being the covariances between variables. The variance–covariance matrix can then be used to solve for its eigenvectors (v) and their corresponding eigenvalues (λ) by finding its spectral decomposition, i.e.

$$\boldsymbol{S} = \boldsymbol{VEV}^\mathrm{T} \tag{2}$$

By the definition of spectral decomposition, the columns of the orthogonal matrix $\boldsymbol{V}$ are the eigenvalues of $\boldsymbol{S}$ and $\boldsymbol{E}$ is a diagonal matrix with its diagonal entries being the corresponding eigenvalues in descending order. This is realized by multiplying $\boldsymbol{V}$ on the right in Eq. (2), which results in $\boldsymbol{SV} = \boldsymbol{VEV}^\mathrm{T}\boldsymbol{V} = \boldsymbol{VE}$, and by looking at the columns of $\boldsymbol{V}$ and diagonal entries of $\boldsymbol{E}$, $\boldsymbol{S}v_i = \lambda_i v_i$. The p eigenvectors resulting from the spectral decomposition are orthogonal and thereby linearly independent and form a p-dimensional space. Let the reduced k linear projections be $\boldsymbol{\xi}$, with $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_k]$.

$$\xi_k = \boldsymbol{v}_k^\mathrm{T}\boldsymbol{x}_c = v_{k1}x_1 + v_{k2}x_2 + \cdots + v_{kp}x_p \tag{3}$$

The variance (var)–covariance matrix of the transformed data $\boldsymbol{\xi}$ is the diagonal eigenvalue matrix.

$$\mathrm{var}(\boldsymbol{\xi}) = \mathrm{var}(\boldsymbol{V}^\mathrm{T}\boldsymbol{x}_\mathrm{c}) = \boldsymbol{V}^\mathrm{T}\,\mathrm{var}(\boldsymbol{x}_\mathrm{c})\boldsymbol{V} = \boldsymbol{V}^\mathrm{T}\boldsymbol{VEV}^\mathrm{T}\boldsymbol{V} = \boldsymbol{E} \tag{4}$$

Therefore, it is evident that the k components of $\boldsymbol{\xi}$ are uncorrelated, and their variances are the eigenvalues. The proportion of variance explained by the reduced k-dimensional principal components is given by $\mathrm{Var}_\mathrm{exp} = \dfrac{\sum_i^k \lambda_i}{\sum_i^n \lambda_i}$.

2.1.2. Singular-value decomposition

A unique SVD exists for any real matrix $\boldsymbol{X} \in \mathbf{R}^{n \times p}$:

$$\boldsymbol{X} = \boldsymbol{UDV}^\mathrm{T} \tag{5}$$

where $\boldsymbol{U} \in \mathbf{R}^{n \times n}$ and $\boldsymbol{V} \in \mathbf{R}^{p \times p}$ are orthogonal matrices. The columns of U are called the left singular vectors and those of V the right singular vectors. $\boldsymbol{D} \in \mathbf{R}^{n \times p}$ has positive singular values only for its diagonal entries, and the number of diagonal entries is equal to rank ($\boldsymbol{X}$). It is generally assumed in this paper that $n > p$ holds for the data matrix $\boldsymbol{X}$. The SVD of $\boldsymbol{X}$ is associated with the spectral decompositions of the matrices $\boldsymbol{X}^\mathrm{T}\boldsymbol{X}$ and $\boldsymbol{XX}^\mathrm{T}$.

The right singular vectors V are the eigenvectors of the matrix $\boldsymbol{X}^\mathrm{T}\boldsymbol{X}$, as shown in Eq. (6).

$$\begin{cases} \boldsymbol{X}^\mathrm{T}\boldsymbol{X} = \boldsymbol{VD}^\mathrm{T}\boldsymbol{U}^\mathrm{T}\boldsymbol{UDV}^\mathrm{T} = \boldsymbol{V}(\boldsymbol{D}^\mathrm{T}\boldsymbol{D})\boldsymbol{V}^\mathrm{T} \\ \boldsymbol{X}^\mathrm{T}\boldsymbol{XV} = \boldsymbol{V}(\boldsymbol{D}^\mathrm{T}\boldsymbol{D})_{p \times p} \end{cases} \tag{6}$$

The left singular vectors U are the eigenvectors of the matrix $\boldsymbol{XX}^\mathrm{T}$ as shown in Eq. (7).

$$\begin{cases} \boldsymbol{XX}^\mathrm{T} = \boldsymbol{UDV}^\mathrm{T}\boldsymbol{VD}^\mathrm{T}\boldsymbol{U}^\mathrm{T} = \boldsymbol{U}(\boldsymbol{DD}^\mathrm{T})\boldsymbol{U}^\mathrm{T} \\ \boldsymbol{XX}^\mathrm{T}\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{D}^\mathrm{T}\boldsymbol{D})_{n \times n} \end{cases} \tag{7}$$

Eq. (8) shows the SVD of the data matrix X when it has a full column rank with $n > p$, where u_i, $i \in \{1, 2, \cdots, n\}$ is the ith column of the matrix of the left singular vectors $\boldsymbol{U}$ and $v_j^{\mathrm{T}}, and\ j \in \{1, 2, \cdots, p\}$ is the jth row of the matrix of right singular vectors V. The upper partition of matrix D is a p by p diagonal matrix of singular values in descending order and the lower partition is a $(n - p)$ by p matrix of zeros.

$$\boldsymbol{X} = \boldsymbol{UDV}^{\mathrm{T}} = [u_1, u_2, \cdots, u_n] \begin{bmatrix} \sigma_1 & 0 & 0 & 0 & \cdots \\ 0 & \sigma_2 & 0 & 0 & \cdots \\ 0 & 0 & \sigma_3 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \cdots \\ 0 & \vdots & \vdots & \vdots & \sigma_p \\ 0 & \vdots & \vdots & \vdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix} \begin{bmatrix} v_1^{\mathrm{T}} \\ v_2^{\mathrm{T}} \\ \vdots \\ v_p^{\mathrm{T}} \end{bmatrix} =$$

$$\sigma_1 u_1 v_1^{\mathrm{T}} + \sigma_2 u_2 v_2^{\mathrm{T}} \cdots + \sigma_p u_p v_p^{\mathrm{T}} = \sum_i^p \sigma_i u_i v_i^{\mathrm{T}} \tag{8}$$

The resulting p rank-1 matrices have linearly dependent rows and columns, which represent the principal components in descending order of importance.

Computationally, SVD and spectral decomposition are similar. SVD can be computed using the QR-SVD algorithm whereas spectral decomposition can be performed using the symmetric QR algorithm. Both algorithms are based on orthogonal similarity transformations that preserve the eigenvalues. The algorithms are iterative for eigenvalue problems with p greater or equal to five, and no general formula exists for the roots of the characteristic polynomial. In this case, the symmetric QR algorithm converges faster. However, SVD is more numerically stable as the explicit formation of the variance–covariance matrix unnecessarily enlarges the condition number of the problem.

2.2. Stockpile simulator

In this paper, we examine the effects of the chevron and windrow stacking methods for a variety of stockpile configurations. Simply put, the chevron stacking method is performed by stacking materials horizontally in one direction followed by stacking another layer of material on top in the opposite direction. In the windrow stacking method, materials are stacked in parallel rows with triangular cross-sections and then more rows are stacked on top between the gaps on the multiple peaks. Chevron stacking tends to lead to particle segregation, whereas the windrow method does not; it reduces fluctuations in the particle size distribution by traversing the stacker much more frequently [9]. Due to the complexity of actual blending operations, it is very difficult, if not impossible, to build a model that perfectly replicates their effects. Hence, we built relatively simple linear block models in MATLAB to simulate the effects of the chevron and windrow blending methods. Like the simulator developed by Marques and Costa in 2013, the stockpile simulator we used in this study is essentially a homogenization simulator for linear cuboid stockpiles [16].

The input to this simulator is a series of predefined mining sequences, and the output is the blocks re-arranged by the algorithm. The simulated stockpiles are defined by three parameters, namely the stockpile height (h), length (l) and width (w). The stockpile capacity can be expressed as $\text{Capacity} = h \times l \times w$. We simulated chevron stockpiles by laying down blocks along the direction of the stockpile length until reaching the predefined stockpile length (l), then laying more blocks on the next level up in the opposite direction. We set the widths of the simulated chevron stockpiles to 1. In the case of the windrow stockpiles, blocks are laid down in the direction of the stockpile length and when the row is filled (stockpile length is reached) another row is added in the stockpile width direction, but the blocks in the row are laid down in a direction opposite to the previous row. This process is repeated until the stockpile width is reached, after which more rows of blocks are stacked on top but with the rows and blocks in the row laid down in opposite directions. In other words, the direction in which the blocks are laid down reverses with each increment of stockpile width, and direction in which rows of blocks are put down reverse with each increment of stockpile height. This process is repeated until the desired stockpile height is reached. The reclaiming process is simulated by taking the average grades of all the blocks in the same reclaiming slice, i.e., all blocks with the same stockpile length (l) value. In other terms, each reclaiming slice has $h \times w$ number of blocks, and the stockpile has a total of h layers with each layer having $l \times w$ blocks.

We evaluated the effect of the blending process primarily using the VRR, which is given by the following [7]:

$$\text{VRR} = \frac{\sigma_{\text{out}}^2}{\sigma_{\text{in}}^2} \tag{9}$$

where σ_{out}^2 and σ_{in}^2 are, respectively, the output and input variances. It is of paramount importance that the VRR is calculated based on the same weight or volume, and in the case of this paper, the number of blocks of material. Since the output of the simulation takes the average grade of all blocks within the same reclaiming slice, the mean of the same number of blocks is calculated while finding the input variance.

We tested multiple different stockpile configurations for the two stacking methods and calculated the VRR for each variable in each configuration scenario. Figs. 1 and 2 show illustrations of the chevron and windrow stockpiles, respectively.

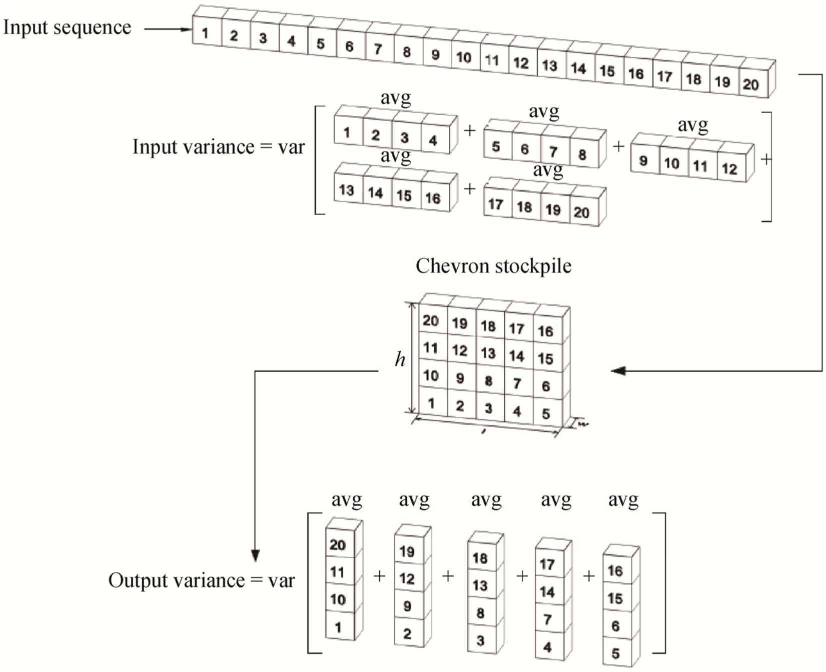

Fig. 1. VRR calculation of a Chevron stockpile.

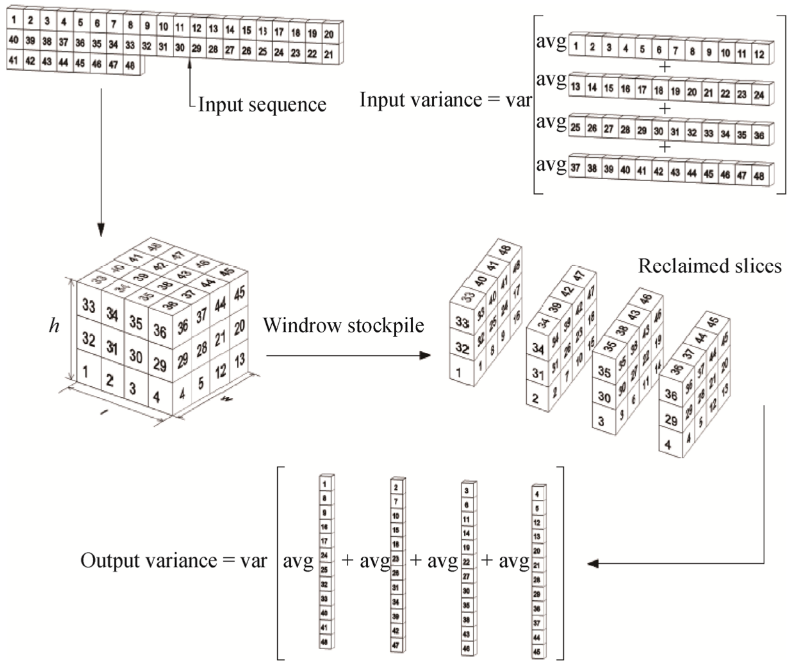

Fig. 2. VRR calculation of a Windrow stockpile.

3. Case study

3.1. Input data and PCA

In this case study, the data input totaled 15000 blocks containing grade information for iron, silica, alumina, and lime. We briefly analyzed the input data and ran them through the PCA algorithm.

Fig. 3 shows a scatterplot matrix of the input data, with the lower-left panels being scatterplots, the diagonal panels being histograms of each variable, and the upper-right panels being the correlations between variables. In this matrix, we can see that the input variables, i.e. the mineral grades, have very complex relationships with each other and are highly correlated, except for lime.

Using the SVD method, we conducted PCA of the dataset using the prcomp() function in R software, with the data matrix centered and scaled. Table 1 shows the proportion of variance explained by each of the principal components. Since PC1 (Principal component 1) alone accounts for just 54.3% of the original variation in the dataset, we used the first two principal components to preserve approximately 80% of the variation.

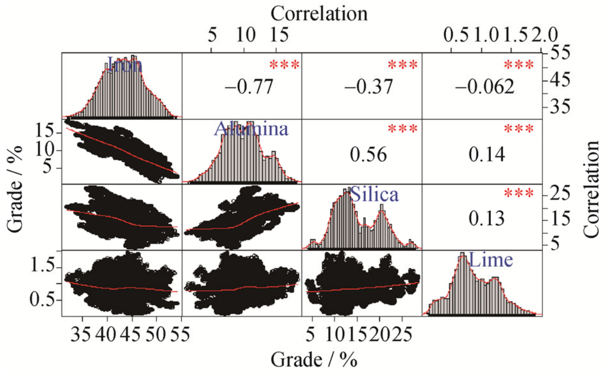

Fig. 3. Scatterplot matrix of case-study input data.

Table 1. Importance of principal components

Principal component	Standard deviation	Proportion of variance	Cumulative proportion
1	1.474	0.543	0.543
2	0.990	0.245	0.788
3	0.807	0.163	0.951
4	0.443	0.049	1.000

Table 2 shows the principal component loadings, which are essentially eigenvectors sorted with respect to their corresponding eigenvalues in descending order. The principal components are linear combinations of the original variables and the loadings represent their relative coefficients. In other words, variables that have large loadings contribute more to a certain principal component. In the case of this dataset, iron and alumina are the primary contributors to PC1, whereas lime contributes overwhelmingly to PC2.

Fundamentally, Table 2 reveals that the two principal components used to reconstruct the data have forms as shown in Eqs. (10) and (11), where the name of the minerals represent the normalized percentage grade.

$$\mathrm{PC1} = 0.576 \times \mathrm{Iron} - 0.631 \times \mathrm{Alumina} - 0.496 \times \mathrm{Silica} - 0.158 \times \mathrm{Lime} \tag{10}$$

$$\mathrm{PC2} = 0.2 \times \mathrm{Iron} - 0.09 \times \mathrm{Alumina} + 0.036 \times \mathrm{Silica} + 0.975 \times \mathrm{Lime} \tag{11}$$

Table 2. Principal component loadings

Principal component	Iron	Alumina	Silica	Lime
1	0.576	−0.631	−0.496	−0.158
2	0.200	−0.090	0.036	0.975
3	0.506	−0.156	0.835	−0.149
4	0.610	0.755	−0.237	−0.046

Fig. 4 shows a biplot of the principal component scores, i.e. the transformed/reduced data. The *x*- and *y*-axes represent standardized PC1 and PC2 scores, and as shown on the axis labels, they account for 54.3% and 24.5% of variation in the data respectively. The four vectors are the transformed variables, which are essentially original variables rebuilt using the chosen principal components. The quality of the representation of each vector by the two chosen PCs is indicated in different colors based on their respective squared cosine values. For any given variable, the sum of the squared cosines from all the PCs should be equal to one. Since the reduced data consist of just two PCs, the better a variable is represented by these two PCs, the closer it is to the circumference of the circle [17]. For this dataset, the first two principal components represent lime, alumina, and iron fairly well, but some of the information from silica is lost in the transformation, as the reconstructed vector has a relatively low cos2 value, which is the 2-norm of the corresponding loading vector. The correlations between variables are largely preserved, as indicated by the angles between vectors.

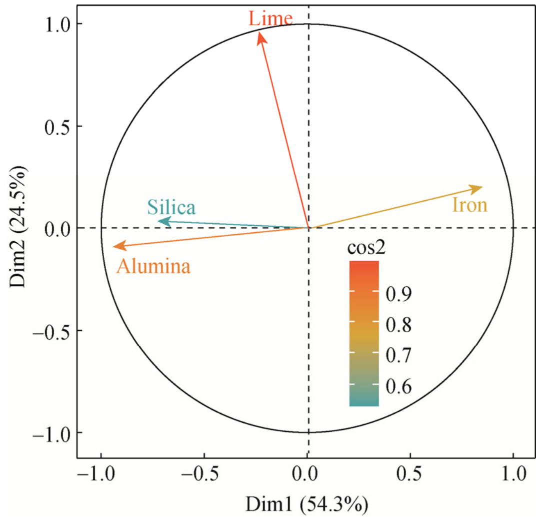

Fig. 4. Principal component biplot.

3.2. Output analysis

We generated 45 stockpile scenarios in total, 15 of which were chevron and the rest windrow, and we created half of the windrow scenarios by switching the values of the stockpile height and width. For all scenarios, we kept the stockpile capacity constant at 15000 blocks. After running the input data and the principal components through the stockpile simulator, we obtained the VRR values shown in Tables 3 and 4.

In the windrow scenarios, we can see that simply switching the width and height values does not change the VRR values at all. Moreover, for both windrow and chevron stacking, the VRR is generally minimized by reducing the stockpile length, which is equivalent to increasing the number of blocks in each reclaiming slice.

Table 3. Chevron stockpile output

Height	Length	Width	VRR_{Iron}	VRR_{Silica}	$VRR_{Alumina}$	VRR_{Lime}	VRR_{PC1}	VRR_{PC2}
5	3000	1	0.0575	0.0574	0.1134	0.1517	0.0719	0.1403
6	2500	1	0.1480	0.1565	0.0425	0.2959	0.1126	0.3209
8	1875	1	0.1960	0.0813	0.0970	0.0555	0.1486	0.0580
10	1500	1	0.0192	0.0231	0.0098	0.0874	0.0152	0.0858
12	1250	1	0.0636	0.1281	0.0189	0.0239	0.0759	0.0263
15	1000	1	0.0200	0.0205	0.0091	0.0080	0.0122	0.0079
20	750	1	0.0060	0.0090	0.0037	0.0085	0.0049	0.0098
24	625	1	0.0169	0.0142	0.0088	0.0183	0.0088	0.0216
25	600	1	0.0101	0.0074	0.0127	0.0230	0.0072	0.0228
30	500	1	0.0047	0.0073	0.0015	0.0037	0.0049	0.0020
40	375	1	0.0018	0.0010	0.0014	0.0041	0.0009	0.0048
50	300	1	0.0074	0.0054	0.0015	0.0045	0.0048	0.0073
60	250	1	0.0008	0.0008	0.0003	0.0005	0.0008	0.0004
75	200	1	0.0011	0.0013	0.0006	0.0006	0.0012	0.0005
100	150	1	0.0020	0.0018	0.0010	0.0003	0.0020	0.0006

Table 4. Windrow stockpile output

Height	Length	Width	VRR_{Iron}	VRR_{Silica}	$VRR_{Alumina}$	VRR_{Lime}	VRR_{PC1}	VRR_{PC2}
1	3000	5	0.0575	0.0574	0.1134	0.1517	0.0719	0.1403
5	3000	1	0.0575	0.0574	0.1134	0.1517	0.0719	0.1403
2	2500	3	0.1480	0.1565	0.0425	0.2959	0.1126	0.3209
3	2500	2	0.1480	0.1565	0.0425	0.2959	0.1126	0.3209
2	1875	4	0.1960	0.0813	0.0970	0.0555	0.1486	0.0580
4	1875	2	0.1960	0.0813	0.0970	0.0555	0.1486	0.0580
2	1500	5	0.0192	0.0231	0.0098	0.0874	0.0152	0.0858
5	1500	2	0.0192	0.0231	0.0098	0.0874	0.0152	0.0858
2	1250	6	0.0636	0.1281	0.0189	0.0239	0.0759	0.0263
6	1250	2	0.0636	0.1281	0.0189	0.0239	0.0759	0.0263
3	1000	5	0.0200	0.0205	0.0091	0.0080	0.0122	0.0079
5	1000	3	0.0200	0.0205	0.0091	0.0080	0.0122	0.0079
4	750	5	0.0060	0.0090	0.0037	0.0085	0.0049	0.0098
5	750	4	0.0060	0.0090	0.0037	0.0085	0.0049	0.0098
3	625	8	0.0169	0.0142	0.0088	0.0183	0.0088	0.0216
8	625	3	0.0169	0.0142	0.0088	0.0183	0.0088	0.0216
5	600	5	0.0101	0.0074	0.0127	0.0230	0.0072	0.0228
5	600	5	0.0101	0.0074	0.0127	0.0230	0.0072	0.0228
3	500	10	0.0047	0.0073	0.0015	0.0037	0.0049	0.0020
10	500	3	0.0047	0.0073	0.0015	0.0037	0.0049	0.0020
4	375	10	0.0018	0.0010	0.0014	0.0041	0.0009	0.0048
10	375	4	0.0018	0.0010	0.0014	0.0041	0.0009	0.0048
2	300	25	0.0074	0.0054	0.0015	0.0045	0.0048	0.0073
25	300	2	0.0074	0.0054	0.0015	0.0045	0.0048	0.0073
3	250	20	0.0008	0.0008	0.0003	0.0005	0.0008	0.0004
20	250	3	0.0008	0.0008	0.0003	0.0005	0.0008	0.0004
3	200	25	0.0011	0.0013	0.0006	0.0006	0.0012	0.0005
25	200	3	0.0011	0.0013	0.0006	0.0006	0.0012	0.0005
5	150	20	0.0020	0.0018	0.0010	0.0003	0.0020	0.0006
20	150	5	0.0020	0.0018	0.0010	0.0003	0.0020	0.0006

3.3. Autocorrelation and effectiveness of blending

The effectiveness of blending operations for this particular dataset is very high with generally low VRR values, as the data is strongly autocorrelated, as shown in Fig. 5. For the number of lags, we chose one tenth the size of the dataset, which is 1500. In the figure, we can see that there is a significant autocorrelation for all four variables, far exceeding the 95% quantile for noise.

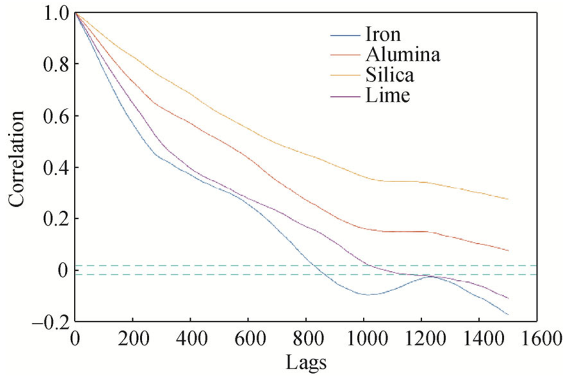

Fig. 5. Autocorrelation plot for original dataset.

Next, we generated an alternative simulated dataset using the Monte Carlo method in **R**, removing autocorrelation while preserving the correlation between variables. Fig. 6 shows a scatterplot matrix of the simulated data, and Fig. 7 shows the autocorrelation of the variables in the simulated data.

The variables of the simulated data exhibit no autocorrelation as the plot follows no obvious pattern and lies almost entirely within the 95% noise region. We ran the simulated data through the same stockpile simulator with identical stockpile configurations, the results of which are shown in Tables 5 and 6.

As shown in the tables, the VRR values for the simulated scenarios all approximate 1, which means that for a dataset without autocorrelation, the effects of blending operations are insignificant.

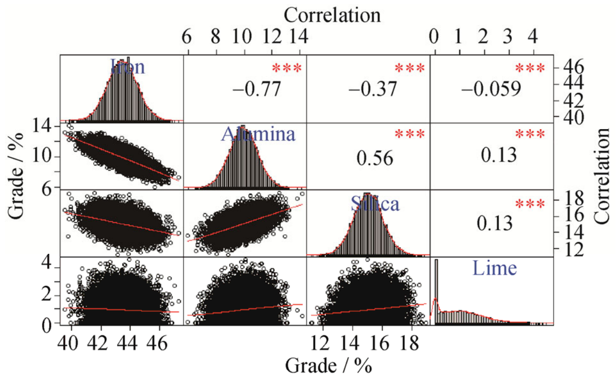

Fig. 6. Scatterplot matrix of simulated data.

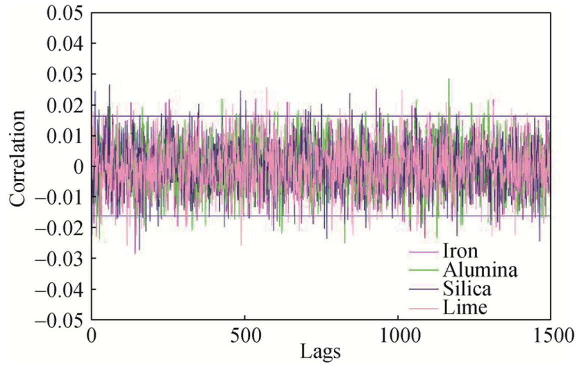

Fig. 7. Autocorrelation plot of simulated data.

Table 5. Chevron stockpile output—simulated data

Height	Length	Width	VRR_{Iron}	$VRR_{Alumina}$	VRR_{Silica}	VRR_{Lime}	VRR_{PC1}	VRR_{PC2}
5	3000	1	1.015	1.028	1.093	0.965	1.052	0.970
6	2500	1	0.995	1.013	1.052	0.950	1.020	0.961
8	1875	1	1.028	1.014	1.052	0.913	1.011	0.931
10	1500	1	0.982	0.979	1.121	0.909	1.020	0.914
12	1250	1	0.996	1.034	1.086	0.924	1.032	0.923
15	1000	1	1.067	0.986	1.140	0.907	1.058	0.913
20	750	1	1.026	0.972	1.103	0.851	1.017	0.877
24	625	1	1.071	1.120	1.169	0.915	1.110	0.953
25	600	1	1.265	1.115	1.148	0.849	1.186	0.847
30	500	1	1.076	0.956	1.130	0.813	1.013	0.836
40	375	1	1.176	1.003	1.144	0.865	1.088	0.919
50	300	1	1.225	1.123	0.962	0.818	1.103	0.841
60	250	1	1.216	1.004	0.994	0.851	1.016	0.932
75	200	1	1.679	1.188	1.092	0.864	1.317	0.871
100	150	1	1.561	1.443	1.003	0.741	1.375	0.809

Table 6. Windrow stockpile output—simulated data

Height	Length	Width	$\mathrm{VRR}_{\mathrm{Iron}}$	$\mathrm{VRR}_{\mathrm{Alumina}}$	$\mathrm{VRR}_{\mathrm{Silica}}$	$\mathrm{VRR}_{\mathrm{Lime}}$	$\mathrm{VRR}_{\mathrm{PC1}}$	$\mathrm{VRR}_{\mathrm{PC2}}$
1	3000	5	1.015	1.028	1.093	0.965	1.052	0.970
5	3000	1	1.015	1.028	1.093	0.965	1.052	0.970
2	2500	3	0.995	1.013	1.052	0.950	1.020	0.961
3	2500	2	0.995	1.013	1.052	0.950	1.020	0.961
2	1875	4	1.028	1.014	1.052	0.913	1.011	0.931
4	1875	2	1.028	1.014	1.052	0.913	1.011	0.931
2	1500	5	0.982	0.979	1.121	0.909	1.020	0.914
5	1500	2	0.982	0.979	1.121	0.909	1.020	0.914
2	1250	6	0.996	1.034	1.086	0.924	1.032	0.923
6	1250	2	0.996	1.034	1.086	0.924	1.032	0.923
3	1000	5	1.067	0.986	1.140	0.907	1.058	0.913
5	1000	3	1.067	0.986	1.140	0.907	1.058	0.913
4	750	5	1.026	0.972	1.103	0.851	1.017	0.877
5	750	4	1.026	0.972	1.103	0.851	1.017	0.877
3	625	8	1.071	1.120	1.169	0.915	1.110	0.953
8	625	3	1.071	1.120	1.169	0.915	1.110	0.953
5	600	5	1.265	1.115	1.148	0.849	1.186	0.847
5	600	5	1.265	1.115	1.148	0.849	1.186	0.847
3	500	10	1.076	0.956	1.130	0.813	1.013	0.836
10	500	3	1.076	0.956	1.130	0.813	1.013	0.836
4	375	10	1.176	1.003	1.144	0.865	1.088	0.919
10	375	4	1.176	1.003	1.144	0.865	1.088	0.919
2	300	25	1.225	1.123	0.962	0.818	1.103	0.841
25	300	2	1.225	1.123	0.962	0.818	1.103	0.841
3	250	20	1.216	1.004	0.994	0.851	1.016	0.932
20	250	3	1.216	1.004	0.994	0.851	1.016	0.932
3	200	25	1.679	1.188	1.092	0.864	1.317	0.871
25	200	3	1.679	1.188	1.092	0.864	1.317	0.871
5	150	20	1.561	1.443	1.003	0.741	1.375	0.809
20	150	5	1.561	1.443	1.003	0.741	1.375	0.809

3.4. Regression analysis

We used multiple regression to identify the relationships between the VRRs of the input materials and the design parameters of the stockpile and used stepwise regression to choose regressors that best describe the models. The possible predictor variables are the stockpile length, width, height, and iswindrow (a binary factor variable that equals 0 if the chevron stockpile is used, 1 otherwise), as well as all their first-order interactions and the second-order terms of stockpile length, width and height. For each response variable, we forwardly selected a model from an initial model with intercepts only, backwardly eliminated another model from an initial model with all possible predictors, and selected a third and final model stepwise that initially consists of only the four main effects. The variable selection criterion is based on Akaike's information criterion (AIC), which measures the closeness between the sample fit and true model fit, where the relative closeness is defined as the Kullback–Leibler divergence from the true model [18]. The AIC can be calculated as follows: $\mathrm{AIC} = -2(\text{Maximum loglikelihood} - \text{Number of parameters})$, and models with lower AIC values are generally preferred. We performed this process using R software with the stepAIC() function. Table 7 shows an illustration of the process of finding the best model for $\mathrm{VRR}_{\mathrm{Iron}}$. Table 8 shows the final results for all the response variables.

Table 7. Stepwise VRR_{Iron} model selection—forward selection only

Step	Step action	Degrees of freedom	Deviance	Residual degrees of freedom	Residual deviance	AIC
1	—	—	—	44	0.145221	−256.13
2	+ length	1	0.07402	43	0.071201	−286.20
3	+ length^2	1	0.008424	42	0.062777	−289.87

Note: Initial model: vrr_iron ~ 1; Final model: vrr_iron ~ length + length^2.

Table 8. Stepwise VRR_{Iron} model selection—forward and backward selection

Step	Step action	Degrees of freedom	Deviance	Residual degrees of freedom	Residual deviance	AIC
1	—	—	—	40	0.071188	−280.21
2	+ length^2	1	0.012905	39	0.058283	−287.21
3	+ height: width	1	0.006292	38	0.051991	−290.35
4	– iswindrow	1	1.03×10^{-5}	39	0.052002	−292.34

Note: Initial model: vrr_iron ~ height + length + width + iswindrow; Final model: vrr_iron ~ height + length + width + length^2 + height × width.

Table 9. Stepwise VRR_{Iron} model selection—backward selection only

Step	Step action	Degrees of freedom	Deviance	Residual degrees of freedom	Residual deviance	AIC
1	—	—	—	32	0.039573	−290.63
2	– width × iswindrow	0	0	32	0.039573	−290.63
3	– width^2	1	0.000717	33	0.040291	−291.82
4	– height × width	1	0.001334	34	0.041625	−292.36
5	– height^2	1	0.001859	35	0.043484	−292.39

Note: Initial model: vrr_iron ~ (height + length + width + iswindrow)^2; Final model: vrr_iron ~ height + length + width + iswindrow + length^2 + height × length + height × iswindrow + length × width + length × iswindrow.

Table 10. Regression results

Model	Adjusted R-squared	AIC
$\text{VRR}_{\text{Iron}} = 1.294\times10^{-3}\times\text{height} + 2.384\times10^{-4}\times\text{length} +$ $5.875\times10^{-3}\times\text{width} - 3.319\times10^{-1}\times\text{iswindrow} -$ $4.835\times10^{-8}\times\text{length}^2 - 1.913\times10^{-5}\times(\text{height}\times\text{length}) +$ $4.681\times10^{-3}\times(\text{height}\times\text{iswindrow}) + 6.723\times10^{-5}\times(\text{length}\times\text{iswindrow}) -$ $1.913\times10^{-5}\times(\text{length}\times\text{width}) + 1.412\times10^{-1}$	0.623	−292.4
$\text{VRR}_{\text{Alumina}} = 1.651\times10^{-4}\times\text{height} + 9.612\times10^{-5}\times\text{length} - 1.811\times10^{-8}\times\text{length}^2 - 3.269\times10^{-2}$	0.57	−357.0
$\text{VRR}_{\text{Silica}} = 1.171\times10^{-8}\times\text{length}^2 - 1.571\times10^{-3}$	0.75	−361.4
$\text{VRR}_{\text{Lime}} = -6.454\times10^{-4}\times\text{height} + 1.187\times10^{-4}\times\text{length} +$ $4.283\times10^{-3}\times\text{width} - 3.319\times10^{-1}\times\text{iswindrow} -$ $2.693\times10^{-5}\times(\text{height}\times\text{length}) + 3.638\times10^{-3}\times(\text{height}\times\text{iswindrow}) +$ $9.391\times10^{-5}\times(\text{length}\times\text{iswindrow}) - 2.693\times10^{-5}\times(\text{length}\times\text{width}) + 3.338\times10^{-1}$	0.73	−287.1
$\text{VRR}_{\text{PC1}} = 1.166\times10^{-4}\times\text{height} + 1.585\times10^{-4}\times\text{length} +$ $4.187\times10^{-4}\times\text{width} - 3.229\times10^{-8}\times\text{length}^2 + 8.414\times10^{-4}\times(\text{height}\times\text{width}) - 1.015\times10^{-1}$	0.67	−321.5
$\text{VRR}_{\text{PC2}} = -6.338\times10^{-4}\times\text{height} + 1.240\times10^{-4}\times\text{length} + 4.601\times10^{-3}\times\text{width} -$ $3.840\times10^{-1}\times\text{iswindrow} - 3.122\times10^{-5}\times(\text{height}\times\text{length}) +$ $3.967\times10^{-3}\times(\text{height}\times\text{iswindrow}) + 1.090\times10^{-4}\times(\text{length}\times\text{iswindrow}) -$ $3.122\times10^{-5}\times(\text{length}\times\text{width}) + 3.986\times10^{-1}$	0.69	−277.1

The resulting model for the principal components is similar to but differs from those for the rest of the variables. Optimizing the principal components rather than the original variables will lead to different stockpile design parameters. However, as

PCA retains as much information as possible during the transformation, the design that minimizes the VRR of the PCs is clearly the mathematically optimal design that aims to minimize the variances for all input variables. This effectively addresses the issue of having to assign a weight or importance to each variable.

We note that the interaction term for the stockpile height and width is unique for the $\mathrm{VRR}_{\mathrm{PC1}}$ model but otherwise, minimizing $\mathrm{VRR}_{\mathrm{PC1}}$ and $\mathrm{VRR}_{\mathrm{PC2}}$ is equivalent to minimizing $\sum_i w_i \mathrm{VRR}_i$, where w_i refers to the weight of each variable and should be set to equal to the sum of the factor loadings of the principal components.

4. Conclusions

Principal component analysis (PCA) can be used in conjunction with multiple regression to design and optimize stockpiles when there are multiple types of materials whose output grades must be controlled. The performance and benefit of applying PCA may potentially increase with the number of material-grade variables studied. Input data that are autocorrelated have a significant impact on the performance of the stockpiles, with reduced variance reduction ratios (VRRs) for increased levels of autocorrelation. The multiple regression results of Table 8 have relatively low adjusted R-squared values, which may be due to some of the variance being uniquely determined by the degree of autocorrelation in the block input. Nevertheless, we found that the VRR is generally reduced with an increasing number of reclamation slices (length) and that the performances of the windrow and chevron methods do not differ significantly. However, additional scenarios and data input are needed to better determine the effects of the design parameters on the VRR.

References

[1] P.M. Gy, Sampling of ores and metallurgical products during continuous transport, *Trans. IMM*, 74(1965), p. 165.

[2] P.M. Gy, A new theory of bed-blending derived from the theory of sampling—Development and full-scale experimental check, *Int. J. Miner. Process.*, 8(1981), No. 3, p. 201.

[3] P.M. Gy, Optimizing the operational strategy of a mine-metallurgy or quarry-cement works complex, *Can. Metall. Q.*, 38(1999), No. 3, p. 157.

[4] S. Zhao, T.F. Lu, B. Koch, and A. Hurdsman, Automatic quality estimation in blending using a 3D stockpile management model, *Adv. Eng. Inf.*, 29(2015), No. 3, p. 680.

[5] S. Zhao, T.F. Lu, B. Koch, and A. Hurdsman, 3D stockpile modelling and quality calculation for continuous stockpile management, *Int. J. Miner. Process.*, 140(2015), p. 32.

[6] G.K. Robinson, How much would a blending stockpile reduce variation?, *Chemom. Intell. Lab. Syst.*, 74(2004), No. 1, p. 121.

[7] P.M. Gy, *Sampling of Heterogeneous and Dynamic Material Systems: Theories of Heterogeneity, Sampling and Homogenizing*, Elsevier, Amsterdam, 1992.

[8] P.A. Dowd, The design of a rock homogenizing stockpile, *Miner. Process. UK*, 1989, p. 63.

[9] M. Kumral, Bed blending design incorporating multiple regression modelling and genetic algorithms, *J. South Afr. Inst. Min. Metall.*, 106(2006), No. 3, p. 229.

[10] J. Sreejith and S. Ilangovan, Optimization of wear parameters of binary Al−25Zn and Al−3Cu alloys using design of experiments, *Int. J. Miner. Metall. Mater*, 25(2018), No. 12, p. 1465.

[11] E. Hosseini, F. Rashchi, and A. Ataie, Ti leaching from activated ilmenite–Fe mixture at different milling energy levels, *Int. J. Miner. Metall. Mater.*, 25(2018), No. 11, p. 1263.

[12] M. De Werk, *Trade-off Between Cost and Performance in Chevron Bed-Blending* [Dissertation], McGill University, Montreal, 2017.

[13] D.G. Paterson, M.N. Mushia, and S.D. Mkula, Effects of stockpiling on selected properties of opencast coal mine soils, *S. Afr. J. Plant Soil*, 36(2019), No. 2, p. 101.

[14] N. Li, J.X. Li, H.M. Long, T.J. Chun, G.T. Mu, and Z.W. Yu, Optimization method for iron ore blending based on the sintering basic characteristics of blended ore, [in] *TMS Annual Meeting & Exhibition*, Springer, Cham, 2018, p. 455.

[15] V. Singh, A. Biswas, S.K. Tripathy, S.K. Chatterjee, and T.K. Chakerborthy, Smart ore blending methodology for ferromanganese production process, *Ironmaking Steelmaking*, 43(2016), No. 7, p. 481.

[16] D.M. Marques and J.F.C.L. Costa, An algorithm to simulate ore grade variability in blending and homogenization piles, *Int. J. Miner. Process.*, 120(2013), p. 48.

[17] H. Abdi, and L.J. Williams, Principal component analysis, *Wiley Interdiscip. Rev. Comput. Stat.*, 2(2010), No. 4, p. 433.

[18] A. Agresti, *Foundations of Linear and Generalized Linear Models*, John Wiley & Sons, Inc., Hoboken, 2015.