

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA  
Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais  
(PPGE3M)

PAULO HENRIQUE FARIA

SIMULAÇÃO GEOESTATÍSTICA MULTIVARIADA COM USO DE PPMT:  
Uma Aplicação para Mapeamento de Incertezas.

Porto Alegre  
2021

PAULO HENRIQUE FARIA

SIMULAÇÃO GEOESTATÍSTICA MULTIVARIADA COM USO DE PPMT:  
Uma Aplicação para Mapeamento de Incertezas.

Dissertação submetida ao Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais da Universidade Federal do Rio Grande do Sul, como requisito parcial à obtenção do título de Mestre em Engenharia, modalidade Acadêmica. Área de Concentração: Tecnologia Mineral.

Orientador: Prof. Dr. João Felipe Coimbra Leite Costa

Porto Alegre

2021

Faria, Paulo Henrique  
SIMULAÇÃO GEOESTATÍSTICA MULTIVARIADA COM USO DE  
PPMT: Uma Aplicação para Mapeamento de Incertezas. /  
Paulo Henrique Faria. -- 2021.  
178 f.  
Orientador: João Felipe Coimbra Leite Costa.

Dissertação (Mestrado) -- Universidade Federal do  
Rio Grande do Sul, Escola de Engenharia, Programa de  
Pós-Graduação em Engenharia de Minas, Metalúrgica e de  
Materiais, Porto Alegre, BR-RS, 2021.

1. Simulação multivariada. 2. PPMT. 3. Médias-k. 4.  
Níquel laterítico. I. Coimbra Leite Costa, João  
Felipe, orient. II. Título.

PAULO HENRIQUE FARIA

SIMULAÇÃO GEOESTATÍSTICA MULTIVARIADA COM USO DE PPMT:  
Uma Aplicação para Mapeamento de Incertezas.

Esta dissertação foi analisada e julgada adequada para a obtenção do título de Mestre em Engenharia, área de concentração de Tecnologia Mineral, e aprovada em sua forma final pelo Orientador e pela Banca Examinadora designada pelo Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais da Universidade Federal do Rio Grande do Sul.

Prof. Dr. João Felipe Coimbra Leite Costa

Orientador

Prof. Dr. Afonso Reguly

Coordenador do PPGE3M

Aprovado em: 29/01/2021

BANCA EXAMINADORA

Dr. Fernando Rosa Guimarães – Anglo American \_\_\_\_\_

Prof. Dr. Marcel Antônio Arcari Bassani - UFRGS \_\_\_\_\_

Profa. Dra. Vanessa Cerqueira Koppe - UFRGS \_\_\_\_\_

Dedico este trabalho à  
minha esposa Aline  
Domingues Morais, pela  
compreensão e apoio  
sempre.

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus, por me permitir enxergar as oportunidades que aparecem pelo caminho.

Aos meus pais Diva Elena Gomes e Pedro Paulo Faria, pela construção do meu caráter.

À minha esposa Aline Domingues Moraes, por sua imensa compreensão, apoio e incentivo, que foram fundamentais para eu concluir mais esta etapa.

Ao Professor Dr. João Felipe Coimbra Leite Costa, por criar um ambiente tão grande de integração entre academia e indústria, permitindo que mesmo os alunos distantes geograficamente da universidade possam obter uma formação de qualidade superior.

Aos colegas do LPM, em especial Marcel Bassani, pela atenção e conhecimentos compartilhados.

À mineradora Anglo American Níquel Brasil, por permitir o uso dos dados neste trabalho e apoiar a pós-graduação.

Aos colegas de trabalho da Anglo American, principalmente ao amigo Dr. Michael Harley, por sua mentoria contínua e enorme contribuição na definição do fluxo de trabalho e scripts *bash*.

“O êxito da vida não se mede pelo caminho que você conquistou, mas sim pelas dificuldades que superou pelo caminho.”

Abraham Lincoln

## RESUMO

Estimativas multivariadas de teores em depósitos minerais de geologia complexa e cujo processo de beneficiamento/metalúrgico necessita controlar múltiplas variáveis é um desafio na indústria mineral. Há dificuldade de se manter a correlação das diversas variáveis entre si pelos métodos tradicionais de estimativa e/ou simulação quando se modela cada variável de maneira independente. Quando as relações multivariadas devem ser reproduzidas nos modelos de teores, é necessário o uso de métodos que garantam essas complexas associações. Este trabalho investiga o uso da técnica *Projection Pursuit Multivariate Transform* (PPMT), a qual descorrelaciona totalmente as múltiplas variáveis de interesse, permitindo a simulação condicional de cada variável no espaço transformado de maneira independente. Ao fim, retro-transforma as variáveis simuladas, reproduzindo as correlações iniciais dos dados e permitindo a medição da incerteza considerando todas as variáveis. Para ilustrar a proposta, PPMT foi aplicada a um depósito de níquel laterítico considerando cinco variáveis de teores: níquel, ferro, sílica, magnésio e cálcio. 50 simulações condicionais de cada variável foram feitas e checadas. As realizações retro-transformadas para o espaço real reproduziram os histogramas, variogramas e correlações multivariadas dos dados. Para o cálculo das incertezas, foram gerados, através da técnica de agrupamento médias-k, painéis representativos de lavra equivalentes a duas e quatro semanas de produção. Foram gerados 50 cenários possíveis de lavra para ambos os períodos. As incertezas foram sumarizadas pelo coeficiente de variação (CV) de todas as realizações simuladas para todas as variáveis dentro dos painéis de lavra, para os 50 cenários. Os resultados foram usados para subdividir a área de estudos em categorias de variabilidade. Para o período de duas semanas, os volumes com CV menor que 5% com 90% de confiança foram atribuídos à categoria 1, enquanto CV maior que 5% se enquadraram na categoria 2. Para o período de quatro semanas, volumes com CV menor que 5% com 90% de confiança permanecem como categoria 2, do contrário, categoria 3. Um modelo simulado multivariado que reproduz as relações iniciais dos dados e um método de categorização foram propostos considerando múltiplas variáveis. Os critérios de categorização (i.e., CV de 5% e 90% de confiança) funcionaram adequadamente e podem ser ajustados a depósitos distintos.

Palavras-chave: Simulação multivariada. PPMT. Médias-k. Níquel laterítico.



## ABSTRACT

Multivariate grade estimation in mineral deposits of complex geology and where beneficiation/metallurgical process needs to control multiple variables is a challenge. It is difficult to maintain the correlation between the variables by traditional estimation and/or simulation methods when each variable is modeled independently. When multivariate relationships must be reproduced in grade models, multivariate geostatistical techniques that guarantee these complex associations are required. This work investigates the use of the multivariate transformation Projection Pursuit Multivariate Transform (PPMT), which fully decorrelates the multiple variables of interest, allowing the conditional simulation of each variable in the transformed space independently. Finally, the simulated variables are back-transformed, reproducing the initial correlations of the data and allowing assessing model uncertainty considering all variables of interest. To illustrate the proposal, the PPMT workflow was applied to a nickel laterite deposit considering five variables important to the metallurgical process: nickel, iron, silica, magnesium and calcium grades. 50 conditional simulations of each variable were made and checked. The back-transformed realizations reproduced the histograms, variograms and multivariate relationships of the data. The simulations were performed at a point-support and upscaled to SMU's. To calculate the uncertainties, representative mining panels equivalent to two and four weeks of production were generated using k-means clustering technique. 50 possible mining scenarios were generated for both production periods. Uncertainties were summarized by the coefficient of variation (CV) of all simulated realizations for all variables within the mining panels, for all 50 mining scenarios. The results were used to define classes of mineral resources. For the two-week period volume, the volumes with CV less than 5% at 90% confidence were assigned to category 1, while those with CV greater than 5% were grouped as category 2. For the four-week period volume, the volumes with CV less than 5% at 90% confidence remains as category 2, otherwise category 3. The multivariate simulation workflow that led to models which reproduce the initial data correlations and a categorization methodology were proposed considering the uncertainties of multiple variables. The categorization criteria (i.e., 5% CV at 90% confidence) work adequately and can be adjusted to distinct mineral deposits.

Keywords: Multivariate Simulation. PPMT. K-means. Nickel laterite.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>10</b>
1.1	META E OBJETIVOS .....	11
1.2	METODOLOGIA .....	12
1.3	ORGANIZAÇÃO DA DISSERTAÇÃO.....	14
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b> .....	<b>15</b>
2.1	KRIGAGEM ORDINÁRIA.....	15
2.2	FATORIZAÇÃO .....	18
<b>2.2.1</b>	<b>Principal Componente Analysis - PCA</b> .....	<b>19</b>
<b>2.2.2</b>	<b>Minimum – Maximum Autocorrelation Factors – MAF</b> .....	<b>20</b>
2.3	COSIMULAÇÃO SEQUENCIAL GAUSSIANA .....	21
2.4	PROJECTION PURSUIT MULTIVARIATE TRANSFORM .....	22
<b>2.4.1</b>	<b>Transformação <i>Normal Score</i></b> .....	<b>22</b>
<b>2.4.2</b>	<b><i>Data Sphering</i></b> .....	<b>23</b>
<b>2.4.3</b>	<b><i>Projection Pursuit</i></b> .....	<b>23</b>
<b>2.4.4</b>	<b><i>Critério de Interrupção de Busca</i></b> .....	<b>25</b>
<b>2.4.5</b>	<b><i>Retro-transformação</i></b> .....	<b>26</b>
2.5	CATEGORIZAÇÃO DE RECURSOS MINERAIS .....	27
<b>3</b>	<b>ESTUDO DE CASO</b> .....	<b>30</b>
3.1	CONTEXTO.....	30
3.2	ANÁLISE EXPLORATÓRIA DOS DADOS .....	31
3.3	TRANSFORMAÇÃO PPMT E SIMULAÇÃO.....	35
3.4	ANÁLISE DE INCERTEZA ESPACIAL E CATEGORIZAÇÃO.....	43
<b>4</b>	<b>CONSIDERAÇÕES FINAIS</b> .....	<b>50</b>
4.1	CONCLUSÕES.....	51
4.2	RECOMENDAÇÕES .....	52
	<b>REFERÊNCIAS</b> .....	<b>54</b>
	<b>APÊNDICE A – PREENCHIMENTO DE DADOS FALTANTES PARA FERRO E CÁLCIO</b> .....	<b>58</b>
	<b>APÊNDICE B – CALIBRAÇÃO DOS PARÂMETROS</b> .....	<b>60</b>

<b>APÊNDICE C – E-TYPE VS. KRIGAGEM ORDINÁRIA.....</b>	<b>68</b>
<b>APÊNDICE D – BASH SCRIPTS.....</b>	<b>70</b>
<b>APÊNDICE E – RSTUDIO SCRIPT .....</b>	<b>170</b>
<b>APÊNDICE F – DATAMINE SCRIPT.....</b>	<b>171</b>

## 1 INTRODUÇÃO

Estimativas multivariadas de teores em depósitos minerais são necessárias em muitas jazidas em exploração, cujas variáveis de controle são definidas muitas vezes pelos processos de beneficiamento/metalúrgico. Esses modelos estimados são usados pelo setor de planejamento de mina para produzir um plano que atinja os parâmetros de alimentação adequados para uma boa recuperação do minério alvo.

Nestas situações, é muito comum que tais estimativas sejam feitas por métodos amplamente difundidos, como krigagem ordinária (Matheron, 1963), com a qual cada variável é estimada de maneira independente, ou mesmo por técnicas de simulações sequenciais gaussianas (Isaaks, 1990). Em ambos os casos, os resultados finais obtidos geram modelos, estimados ou simulados, em que as variáveis não mantêm necessariamente as correlações iniciais dos dados, propagando esse problema para as etapas de sequenciamento de lavra e beneficiamento, uma vez que os materiais selecionados para compor a alimentação da planta não representam de maneira correta a relação das variáveis *in situ*.

Alguns métodos de geoestatística multivariada assumem um modelo multigaussiano, com as correlações multivariadas parametrizadas (Journel & Huijbregts, 1978). No entanto, é sabido que a natureza raramente apresenta variáveis com relações multigaussianas (Barnett, 2015) requerendo transformações não lineares dos dados originais.

Nos casos em que manter as correlações entre as variáveis de interesse é importante para a previsibilidade do processo, é recomendado o uso de técnicas que garantam a reprodução das relações multivariadas dos dados, gerando resultados que permitam uma maior previsibilidade do comportamento do minério durante o processamento mineral.

Na geoestatística multivariada, o grau de complexidade aumenta ao se trabalhar com mais de três variáveis correlacionadas, dificultando demasiadamente a adaptação de modelos lineares de correogionalização permissíveis (Journel & Huijbregts, 1978).

Para solucionar esta questão, surge a fatorização, a qual descorrelaciona os dados por meio de transformações lineares, de maneira que as estimativas podem então ser feitas em cada variável de maneira independente por métodos univariados. Tal descorrelação é feita para todas as variáveis simultaneamente, de forma que as

correlações iniciais serão mantidas ao se aplicar a retro-transformação nos dados estimados. Uma restrição deste método é que é exigido isotopia da base de dados, o que muitas vezes não ocorre na prática, mas pode ser resolvido por meio de técnicas de preenchimento de dados (*data imputation*) ou se perder parte da base de dados e trabalhar apenas com um subconjunto de amostras que contenham informações para todas as variáveis.

Alguns métodos de fatorização que permitem a descorrelação dos dados são *Principal Component Analysis* (PCA) (Pearson, 1901) e *Minimum/Maximum Autocorrelation Factors* (MAF) (Switzer & Green, 1984), os quais servem apenas para descorrelacionar dados que possuem correlações lineares, não se aplicando a bancos de dados que possuem correlações mais complexas (Bassani, 2018), de grau mais elevado. Para contornar este problema, foram desenvolvidas técnicas que possibilitam transformações multigaussianas, como a *Stepwise Conditional Transformation*, o qual garante que as variáveis transformadas são todas multigaussianas com correlação zero (Leuangthong & Deutsch, 2003). Estes autores afirmam que este método possui limitações quanto à quantidade de dados, mencionando uma regra geral de  $10^n$  a  $20^n$  dados, onde  $n$  é o número de variáveis em questão, para se ter distribuições bem discretizadas.

Para contornar esta questão e permitir o trabalho com dados de correlações complexas e múltiplas variáveis (mais de quatro), Barnett et al. (2014) propuseram a técnica conhecida por *Projection Pursuit Multivariate Transform* (PPMT), a qual permite a descorrelação e transformação multigaussiana completa dos dados para qualquer número de variáveis. Alguns trabalhos demonstram a aplicabilidade esta técnica para simulação geoestatística multivariada em depósitos de bauxita (Bassani et al. 2018) e minério de ferro (Battalgazy & Madani, 2019). Esta técnica é o foco da discussão e aplicação prática desta dissertação, com estudo de caso aplicado a um depósito de níquel laterítico.

## 1.1 META E OBJETIVOS

A meta desta dissertação é investigar uma metodologia para construção de modelos que reproduzam as relações complexas em depósitos multivariados, no qual os métodos que necessitam da aplicação de um modelo linear de correção regionalização não são suficientes para se atingir resultados satisfatórios.

Os objetivos específicos associados à meta são:

- (i) Avaliar a aplicabilidade do método PPMT como técnica de fatorização para lidar com problemas de estimativas multivariadas complexas e comparar os resultados desta metodologia proposta com o procedimento utilizado em um depósito mineral de níquel laterítico, o qual aplica como padrão geoestatística linear univariada (krigagem ordinária);
- (ii) Investigar o uso do modelo resultante da simulação multivariada para mapeamento da incerteza dos teores e subdividir a área de estudos em diferentes categorias com base na incerteza, tendo como foco o plano de alimentação da planta metalúrgica no curto prazo.

Uma vez que estes objetivos sejam alcançados, torna-se possível a elaboração de planos de lavra mais robustos, baseados em uma medida de variabilidade quantitativa que considera todas as variáveis em estudo de maneira conjunta, acarretando uma melhor previsibilidade na alimentação da planta no curto prazo. Além disso, esta categorização também contribui para melhor definição dos planos de sondagem, permitindo a priorização de novos furos em áreas com alta variabilidade.

## 1.2 METODOLOGIA

Este trabalho foi desenvolvido aplicando a metodologia descrita a seguir a um depósito de níquel laterítico atualmente em produção, pertencente à mineradora Anglo American plc. e situado no município de Barro Alto-GO. As variáveis estudadas foram: Ni, Fe, SiO<sub>2</sub>, MgO e CaO, as quais são as mais importantes para o processo pirometalúrgico.

A metodologia deste trabalho teve como premissa fundamental a criação de um procedimento semi-automatizado por scripts para permitir a implementação rápida e prática do fluxo de trabalho PPMT com geração de vários cenários de simulação. Modificando-se alguns parâmetros de entrada, tais como número mínimo e máximo de amostras e elipsoide de busca, se pode escolher aquele que leve aos melhores resultados em termos de reprodução dos histogramas e variogramas dos dados, assim como representatividade do depósito.

Foram criados scripts em dois ambientes diferentes. As primeiras etapas, incluindo análise exploratória dos dados, *data imputation* (Barnett & Deutsch, 2013),

transformação PPMT, simulação, retro-transformação e validações (histogramas, variogramas e correlações) foram criadas em *bash scripting* (Robbins & Beebe, 2005), as quais são interpretadas no terminal Unix MobaXTerm® (Mobatek) e servem para executar os comandos executáveis do GsLib (Deutsch & Journel, 1998) e também de outros executáveis disponíveis na plataforma CCG (*Centre for Computational Geostatistics – University of Alberta, Canada*) de maneira ordenada e sequencial.

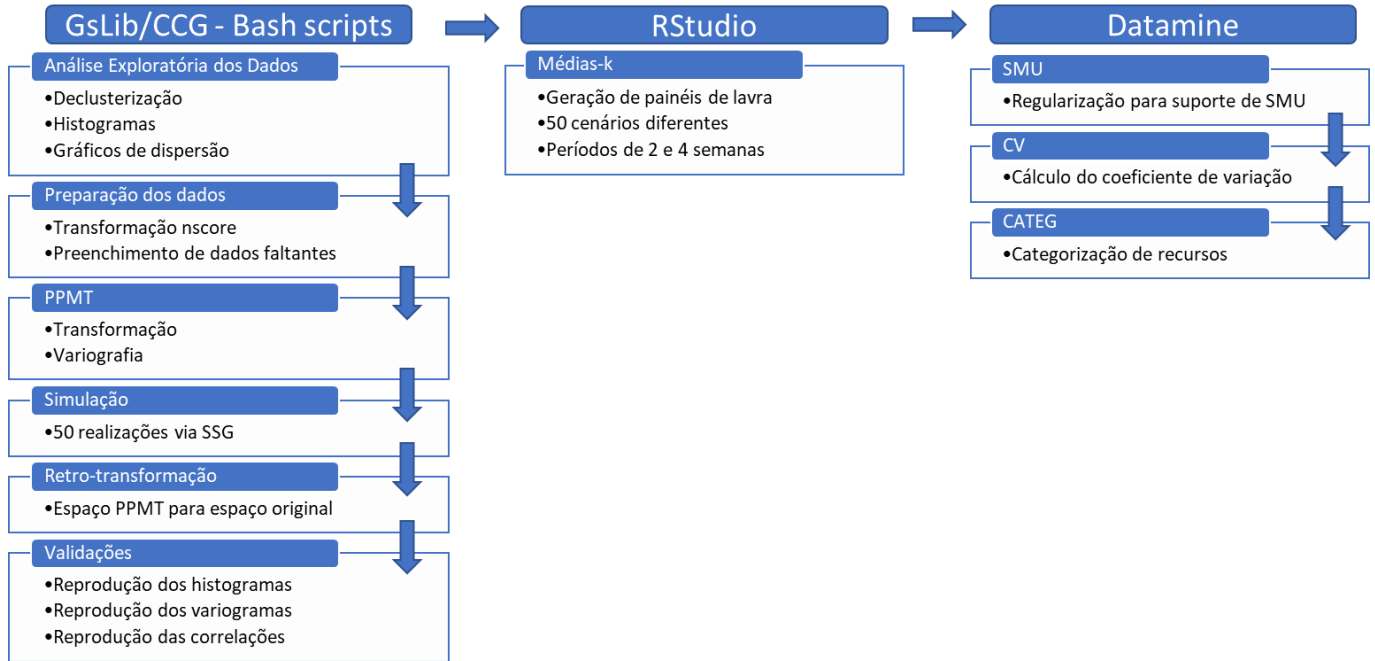
Com os resultados validados na etapa anterior, as próximas etapas foram implementadas em *scripts* via Studio RM® (Datamine Software), incluindo importação das realizações simuladas para a plataforma Datamine, regularização para o suporte de SMU (*Selective Mining Unit*), cálculo do coeficiente de variação (CV), categorização dos recursos nas categorias 1, 2 e 3 (representado baixa, média e alta variabilidade para o curto prazo, respectivamente) e comparações entre as realizações (incluindo a média E-type) com resultados de estimativa por krigagem ordinária.

Para o cálculo da variabilidade por meio do CV, foram gerados painéis representativos de lavra para períodos de duas e quatro semanas, por meio da técnica de agrupamento de médias-k (MacQueen, 1967). Para simular a possibilidade de variação dos polígonos de lavra, a depender do modelo geológico de entrada e/ou da estratégia de lavra, foram gerados 50 cenários possíveis de lavra para ambos os períodos de produção. Esta etapa para geração dos clusters foi executada no software RStudio®.

De maneira geral, o fluxo de trabalho seguiu a seguinte ordem:

- Análise exploratória dos dados;
- Preenchimento de dados faltantes (*data imputation*);
- Aplicação do fluxo de trabalho PPMT (transformação, simulação e retro-transformação);
- Validação dos resultados (reprodução dos histogramas, variogramas e correlações);
- Medição da variabilidade espacial conjunta de todas as variáveis;
- Subdivisão da área de estudo em categorias, com base na análise de incerteza espacial focada no curto prazo.

O fluxograma a seguir exemplifica de maneira gráfica o fluxo de trabalho (Figura 1). Detalhes de cada uma dessas etapas serão apresentados no capítulo 3.



**Figura 1: Fluxograma esquemático do fluxo de trabalho executado.**

### 1.3 ORGANIZAÇÃO DA DISSERTAÇÃO

As próximas seções desta dissertação estão organizadas da seguinte forma:

- O capítulo 2 apresenta uma revisão da bibliografia relacionada ao tema, incluindo aspectos gerais dos métodos mencionados anteriormente que não atingem por completo o objetivo proposto neste trabalho (krigagem ordinária, cosimulação e fatorizações PCA e MAF), além de uma revisão mais detalhada da técnica PPMT, sendo esta o foco deste trabalho;
- O capítulo 3 discorre sobre a implementação do fluxo de trabalho pertinente à dissertação, evidenciando os aspectos práticos aplicados a um caso real de um depósito de níquel laterítico;
- O capítulo 4 apresenta as considerações finais, revisando o atingimento dos objetivos propostos e abordando algumas recomendações para trabalhos futuros relacionados a este assunto.



## 2 REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta uma revisão dos conceitos teóricos relacionados aos métodos de estimativa/simulação quais sejam: krigagem ordinária, fatorizações PCA e MAF, cosimulação, além de uma revisão mais detalhada da técnica PPMT, a qual permite o atingimento da meta desta dissertação a partir da decorrelação e transformação multigaussiana completa dos dados para qualquer número de variáveis.

### 2.1 KRIGAGEM ORDINÁRIA

Krigagem é o nome dado a uma gama de técnicas de estimativa que tem por objetivo minimizar os erros dos valores estimados, comumente pelo método dos mínimos quadrados (Sinclair & Blackwell, 2004). Este processo foi chamado por Georges Matheron (1963) de krigagem, em homenagem ao engenheiro de minas Danie Krige (Clark, 2001), o qual desenvolveu empiricamente trabalhos com médias ponderadas em minas de ouro da África do Sul. Este método de geoestatística desenvolvido por Matheron gerou o que se chama de melhor estimador linear não enviesado; em inglês: "*Best Linear Unbiased Estimator*", referido por vezes apenas pela sigla BLUE. A krigagem é "*linear*" porque suas estimativas são combinações lineares ponderadas dos dados disponíveis; é "*unbiased*" porque produz uma estimativa de forma que o resíduo (ou erro) seja centrado em zero; e é "*best*" porque tem o objetivo de minimizar a variância do erro, ou seja, pouco espalhamento em torno da média zero. Este último ponto é o que distingue a krigagem dos outros métodos lineares (Isaaks & Srivastava, 1989).

Diferentemente dos estimadores clássicos, como por exemplo os métodos do inverso do quadrado da distância (IQD) e do vizinho mais próximo, a krigagem usa a informação da continuidade espacial para realização das estimativas nos locais não amostrados.

O princípio básico de uma estimativa linear de uma variável em um local não amostrado, usando as amostras da vizinhança do local a ser estimado e atribuindo pesos a estas amostras de acordo com sua distribuição espacial, usa a seguinte equação:

$$Z^*(u) = \sum_{i=1}^n \lambda_i z(u_i) \quad (1)$$

sendo  $Z^*(u)$  o valor da variável estimada na posição  $u$ ,  $z(u_i)$  são os valores da variável

distribuídos nas posições  $u_1, \dots, u_n$  e  $\lambda_i$  os pesos atribuídos às  $n$  amostras em cada posição.

Para o caso dos estimadores de krigagem, os mesmos consistem de variações do estimador definido por (Goovaerts, 1997):

$$Z^*(u) - m(u) = \sum_{i=1}^{n(u)} \lambda_i(u) [z(u_i) - m(u)] \quad (2)$$

Nesta equação, tem-se incluída a média dos dados  $m(u)$ , trabalhando desta maneira com uma variável estandardizada, que representa o resíduo (valor lido menos a média). Estes estimadores têm como objetivo a minimização do erro, ou seja, mínima variância do erro ( $\sigma_K^2$ ), sob a condição de não tendenciosidade, ou seja, a média dos resíduos deve ser igual a zero. Estas condições estão representadas, respectivamente, pelas equações 3 e 4:

$$\sigma_K^2(u) = \text{Var}\{Z^*(u) - Z(u)\} \quad (3)$$

$$E\{Z^*(u) - Z(u)\} = 0 \quad (4)$$

Derivando-se a equação (3) a fim de encontrar o valor mínimo da variância e expandindo-se os termos das equações 1 e 2, os pesos a serem usados na equação (2) de forma a atender as condições acima são obtidos pela seguinte relação expressa na forma matricial:

$$\begin{matrix} & C & & \lambda & & C' \\ \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix} & & & \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} & = & \begin{bmatrix} C_{10} \\ C_{20} \\ \vdots \\ C_{n0} \end{bmatrix} \end{matrix} \quad (5)$$

em que a matriz  $C$  representa as covariâncias entre os pares de amostras que serão usadas na estimativa, a matriz  $C'$  representa as covariâncias entre as amostras e o ponto "0" a ser estimado, e a matriz  $\lambda$  representa os pesos de cada amostra. Os valores das covariâncias podem ser facilmente obtidos por meio de variogramas, devido a sua relação dada por:

$$\gamma(h) = \frac{1}{2N(h)} \sum (v_i - v_j)^2 \quad (6)$$

$$C(h) = C(0) - \gamma(h) \quad (7)$$

onde  $C(h)$  é a covariância das amostras separadas de  $h$  metros e  $C(0)$  é a covariância das amostras separadas de zero metros, que corresponde ao *sill* do variograma ( $\gamma(h)$ );  $v_i$  e  $v_j$  são duas amostras separadas de  $h$  metros.

Este sistema de matrizes é chamado de Sistema de Equações de Krigagem Simples e o modelo assume a estacionaridade da média conhecida e igual para toda a área estudada. Porém, em muitas estimativas de dados de ciências da Terra, como, por exemplo, estimativa de teores em mineração, assume-se que a média varie localmente e seja desconhecida, devendo-se aplicar uma forma de krigagem que atribui peso zero à média global e peso igual a 1 para o restante das amostras consideradas na estimativa. Esta é a chamada krigagem ordinária (Matheron, 1971) e é a mais aplicada na indústria mineral.

A krigagem ordinária é representada pela seguinte equação:

$$Z^*(u) = \sum_{i=1}^{n(u)} \lambda_i(u) z(u_i) \quad (8)$$

Este estimador é não tendencioso se a somatória dos pesos ( $\sum_{i=1}^{n(u)} \lambda_i(u)$ ) é igual a 1 e a variância é minimizada através da construção de um sistema de  $n+1$  equações:

$$\begin{matrix} & C & & \lambda & & C' \\ \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} & 1 \\ C_{21} & C_{22} & \dots & C_{2n} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix} & & \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \\ \mu \end{bmatrix} & = & \begin{bmatrix} C_{10} \\ C_{20} \\ \vdots \\ C_{n0} \\ 1 \end{bmatrix} & (9) \end{matrix}$$

Neste sistema, o termo  $\mu$  é o parâmetro de Lagrange, adicionado para permitir a solução única do sistema sob a condição de não enviesamento.

Apesar da krigagem ordinária oferecer a melhor estimativa não enviesada de um local não amostrado, a mesma não garante a reprodução das correlações multivariadas dos dados, uma vez que é executada de maneira independente para cada variável. Há técnicas alternativas de krigagem que possibilitam uma melhora na

correlação entre as variáveis nos valores estimados, como a cokrigagem, método este que leva em conta a continuidade espacial conjunta entre as diferentes variáveis. Porém, para que os sistemas de cokrigagem possuam solução, a modelagem das continuidades espaciais deve respeitar o chamado Modelo Linear de Correogionalização (MLC), o qual não é simples e sua complexidade aumenta com o número de variáveis. O método de cokrigagem não será detalhado neste trabalho, maiores detalhes podem ser encontrados em Goovaerts (1997), Wackernagel (1998) e Isaaks & Srivastava (1989).

## 2.2 FATORIZAÇÃO

Como já mencionado, a fatorização compreende técnicas que permitem descorrelacionar múltiplas variáveis por meio de transformações lineares, permitindo a condução de estimativas independentes de cada variável por métodos univariados. Uma vez que as variáveis são descorrelacionadas em conjunto, as correlações iniciais são mantidas ao se retro-transformar os dados para o espaço real.

Antes de efetuar as transformações, é necessário verificar se o banco de dados é isotópico. Caso não seja (como ocorre na maioria das vezes na prática), é necessário usar técnicas de preenchimento de dados (data imputation) ou assumir a possibilidade de perder parte da base de dados e trabalhar apenas com um subconjunto dos dados contendo informações para todas as variáveis. Isto, porém, além da perda de parte da informação, pode implicar também na geração de viés, caso a falta de dados não seja aleatória (Little & Rubin, 2002). Além disso, também deve ser checado quais variáveis apresentam correlação entre si, pois, apenas tais variáveis serão descorrelacionadas em conjunto. As variáveis que não possuem correlação devem ser estimadas separadamente. Na Tabela 1, tem-se o exemplo de uma matriz de correlação, em que as células destacadas em verde apresentam variáveis com alta correlação, podendo estas serem consideradas na fatorização. Esta decisão deve ser avaliada para atender a especificidade de cada estudo.

A seguir são descritos os métodos PCA e MAF, os quais são relativamente bem difundidos e aplicados em trabalhos que envolvam múltiplas variáveis, servindo apenas para descorrelacionar dados que possuem correlações lineares, não se aplicando à bancos de dados que possuem correlações complexas.

**Tabela 1: Matriz com valores de coeficiente de correlação entre 5 variáveis.**

	U1	U2	U3	U4	U5
U1	1.0	0.6	0.2	0.1	0.3
U2	0.6	1.0	0.4	0.5	0.8
U3	0.2	0.4	1.0	0.7	0.9
U4	0.1	0.5	0.7	1.0	0.2
U5	0.3	0.8	0.9	0.2	1.0

### 2.2.1 Principal Componente Analysis - PCA

Este método foi desenvolvido originalmente por Pearson (1901) e é uma técnica amplamente difundida para o tratamento de dados multivariados. Para variáveis regionalizadas, este método garante a descorrelação para um vetor de separação nulo ( $h=0$ ) (Boezio, 2010). Transformações lineares são aplicadas para transformar um conjunto de variáveis correlacionadas em fatores descorrelacionados, por meio de fatores ortogonais que extraem sucessivamente a máxima parte da variância total dos dados (Wackernagel, 1998).

A equação a seguir representa a combinação linear para obtenção dos componentes  $Z_K$  descorrelacionados:

$$Z_K = \mathbf{a}_K^T \Phi \quad (10)$$

em que  $\mathbf{a}_K^T$  representa o vetor transformação transposto e  $\Phi$  é a função randômica que define as variáveis envolvidas ( $U_1, U_2, \dots, U_N$ ).

Os vetores transformação são obtidos através da decomposição espectral da matriz variância-covariância dos atributos:

$$\mathbf{Cov}_\Phi = \begin{bmatrix} \text{Var}(U_1) & \dots & \text{Cov}(U_1 U_N) \\ \vdots & \ddots & \vdots \\ \text{Cov}(U_N U_1) & \dots & \text{Var}(U_N) \end{bmatrix} \quad (11)$$

A decomposição espectral é realizada definindo-se os autovalores  $\lambda$ , de forma que:

$$\det(\mathbf{Cov}_\Phi - \lambda \mathbf{I}) = 0, \quad \mathbf{I} \text{ é uma matriz identidade.} \quad (12)$$

$$\mathbf{Cov}_\Phi \begin{bmatrix} a_{i1} \\ a_{ij} \\ \vdots \\ a_{iN} \end{bmatrix} = \lambda_i \begin{bmatrix} a_{i1} \\ a_{ij} \\ \vdots \\ a_{iN} \end{bmatrix} \quad (13)$$

A solução dos sistemas acima resulta na obtenção dos autovalores  $a_i$ , os quais definem o vetor transformação. A retro-transformação dos dados estimados é feita multiplicando-se os componentes descorrelacionados pela inversa do vetor transformação transposto:

$$\Phi(u) = (\mathbf{a}^T)^{-1}Z_K(u) \quad (14)$$

Como mencionado, importante ressaltar que este método não elimina a correlação espacial por completo de pontos separados de um vetor  $h$ . A descorrelação ocorre apenas para as distâncias  $h$  iguais a zero. Isso pode ser observado no correlograma dos dados descorrelacionados, em que aparecem valores de correlação diferentes de zero para distâncias também diferentes de zero.

### 2.2.2 Minimum – Maximum Autocorrelation Factors – MAF

Minimum/Maximum Autocorrelation Factors (MAF) foi proposto inicialmente por Switzer & Green (1984) como uma alternativa ao método PCA. Na transformação MAF, se busca maximizar a autocorrelação entre as observações da vizinhança. O conceito básico é que as observações de interesse apresentam elevada autocorrelação, enquanto o ruído presente exibe baixa autocorrelação (Larsen, 2002).

A vantagem da decomposição MAF em relação à fatorização por PCA está no fato de que aquela incorpora a descorrelação espacial para vetores de separação  $h$  maiores que zero (Boezio, 2010).

Assim como no método PCA, a decomposição MAF consiste em transformar um conjunto de variáveis correlacionadas em uma combinação linear de variáveis independentes por meio de um vetor transformação:

$$M_K = \mathbf{a}_K^T \Phi \quad (15)$$

Os fatores MAF são obtidos por meio da decomposição espectral da matriz  $\mathbf{Cov}_{\phi\delta}\mathbf{Cov}_{\phi}^{-1}$ , em que  $\mathbf{Cov}_{\phi}^{-1}$  é o inverso da matriz covariância e  $\mathbf{Cov}_{\phi\delta}$  é a matriz covariância para diferenças de vetores separados por  $h = \delta$ :

$$\mathbf{Cov}_{\phi\delta} = \mathbf{Cov}[(\Phi(u) - \Phi(u + \delta)), (\Phi(u) - \Phi(u + \delta))] = 2\mathbf{\Gamma}_{\phi}(\delta) \quad (16)$$

Sendo  $\mathbf{\Gamma}_{\phi}(\delta)$  a matriz de variogramas para  $h = \delta$ .

A retro-transformação é dada por:

$$\Phi(u) = (\mathbf{a}^T)^{-1}M(u) \quad (17)$$

### 2.3 COSIMULAÇÃO SEQUENCIAL GAUSSIANA

Cosimulação Sequencial Gaussiana (SGCS) é um método que permite a simulação conjunta de múltiplas variáveis, reproduzindo as distribuições e continuidade espacial cruzada das variáveis.

Conforme Verly (1993), para um conjunto de  $K$  variáveis  $[z_k(u), k = 1 \text{ até } K]$  consideradas realizações de  $K$  funções randômicas  $[Z_k(u), k = 1 \text{ até } K]$ , sendo  $u$  a posição no espaço, a cosimulação se propõe a gerar realizações conjuntas das  $K$  funções randômicas de forma que: i) a distribuição e variograma de cada função randômica sejam reproduzidas; ii) os cross-variogramas entre as funções randômicas sejam reproduzidos; iii) os valores amostrados sejam honrados.

Assume-se então que a função randômica  $Z_k(u)$  é estritamente estacionária e que existe uma função randômica  $Y_k(u)$  que corresponde à transformação da função original para o espaço gaussiano (*normal scores*), a qual também possui estacionaridade estrita e distribuição normal (Journel & Huijbregts, 1978). Assume-se também que as distribuições são multigaussianas. Com isso, pode se obter a probabilidade conjunta de  $[Y_k(u), k = 1 \text{ até } K]$  desde que a média, a auto e cross-covariâncias sejam conhecidas. As médias são zero por construção e as covariâncias são dadas pela seguinte correlação:

$$C_{kk'}(h) = C_{kk'}(0) - \gamma_{kk'}, \quad k = 1 \text{ até } K, k' = 1 \text{ até } K \quad (18)$$

Partindo dessas premissas, segue-se com o passo-a-passo da cosimulação sequencial gaussiana: seleciona-se um local aleatório; executa-se uma tiragem aleatória condicionada aos dados (*normal scores*) da vizinhança; incorpora-se o valor simulado ao conjunto de dados; repete o processo até não haver mais locais do grid (nós) a serem simulados.

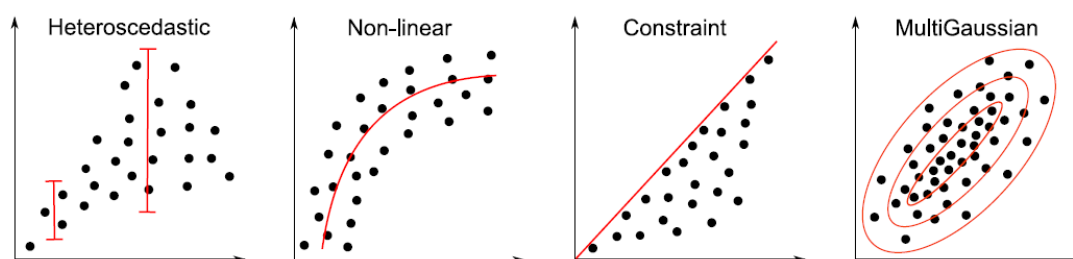
Este método pode ser aplicado sempre que houver correlação entre as variáveis. Contudo, as correlações reproduzidas não necessariamente possuem um significado físico consistente. SGCS pode ser aplicado para mais de duas variáveis correlacionadas, contudo a complexidade dos modelos de correogionalização

umentam significativamente. Outra limitação do método é assumir a multinormalidade, uma vez que esta não pode ser checada e na maioria das vezes não condiz com a realidade prática de variáveis geológicas.

## 2.4 PROJECTION PURSUIT MULTIVARIATE TRANSFORM

Esta técnica, também referida pela sigla PPMT, foi proposta por Barnett *et al.* (2014) para contornar as limitações relacionadas nas técnicas anteriores, permitindo assim a tratativa de correlações multivariadas complexas para qualquer número arbitrário de variáveis. Importante ressaltar que como uma técnica multivariada, ela requer que base de dados seja isotópica.

A Figura 2 mostra alguns exemplos de correlações complexas de dados que podem ser reproduzidas pelo método PPMT.



**Figura 2: Representação esquemática de relações multivariadas complexas, incluindo heterocedasticidade, não linearidade, restrição, as quais contrastam com a forma elíptica da correlação entre variáveis multigaussianas. Fonte: Barnett *et al.* (2014).**

Esta técnica é uma modificação da Projection Pursuit Density Estimation (PPDE) proposta inicialmente por Friedman & Tukey (1974) para endereçar problemas geológicos multidimensionais e com correlações complexas. Embora essas duas técnicas sejam conceitualmente muito parecidas, a PPMT introduziu algumas melhorias por meio de modificações, tais quais: transformação dos dados para o espaço gaussiano como etapa adicional de pré-processamento; utilização de uma abordagem diferente de ortogonalização dos dados (*Data Sphering*); e implementação de um critério para interromper as iterações de busca da projeção. Estas etapas assim como as outras envolvendo a PPMT serão descritas em maior detalhe a seguir.

### 2.4.1 Transformação *Normal Score*

A transformação *normal score* (Deutsch & Journel, 1998) é aplicada a cada uma



das variáveis. Esta é a primeira etapa de pré-processamento antes da aplicação da técnica PPMT propriamente dita. O objetivo desta etapa é reduzir a influência dos *outliers* na etapa seguinte de *Data Sphering*, a qual é sensível a valores extremos e, portanto, pode se beneficiar desta transformação.

Outro ponto observado por Barnett *et al.* (2014) é que o fato de a base de dados estar transformada para o espaço gaussiano contribui para a redução do número de iterações na etapa de *Projection Pursuit*.

#### **2.4.2 Data Sphering**

O procedimento conhecido por *Data Sphering* é similar à técnica PCA e aplica uma série de rotações nos dados a fim de extrair a correlação dos mesmos e ao mesmo tempo deixar os dados com uma variância unitária. Estas características contribuem para a etapa de *Projection Pursuit*.

O *Data Sphering* sugerido por Barnett *et al.* (2014) adequado à técnica PPMT segue a seguinte fórmula:

$$x = S^{-1/2}y, \text{ onde } S^{-1/2} = aD^{-1/2}a^T \quad (19)$$

Na equação acima, a variável transformada  $x$  é uma função dos dados originais  $y$  (já no espaço normal) pela matriz *sphering*  $S^{-1/2}$ . Os termos  $D$  e  $a$  são matrizes de autovalores e autovetores obtidos pela decomposição espectral da matriz covariância.

#### **2.4.3 Projection Pursuit**

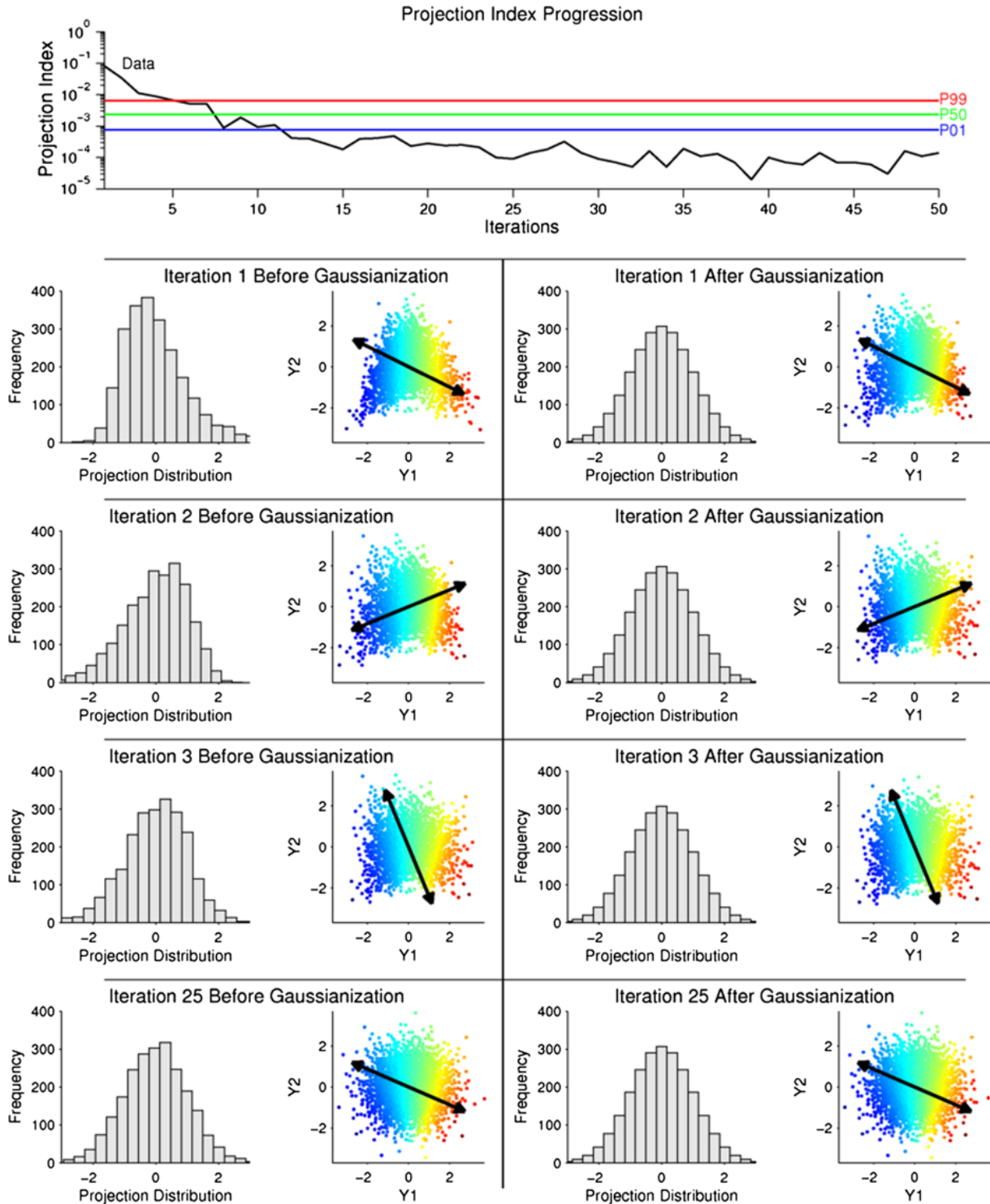
A técnica de *Projection Pursuit*, proposta originalmente por Kruskal (1969, 1972 apud Huber, 1985), tem como objetivo primário investigar características presentes na dispersão de dados multidimensionais por meio da projeção dos mesmos em espaços transformados de dimensões reduzidas, sendo a característica de interesse representada por um índice de projeção (Huber, 1985).

Para a aplicação PPMT, esta etapa de *Projection Pursuit* segue o mesmo princípio aplicado por Friedman & Tukey (1974) e busca gaussianizar a base de dados em todas as dimensões. Para isso, assume-se que a projeção de uma distribuição multigaussiana K-dimensional (onde K é igual ao número de variáveis) também possui

uma distribuição gaussiana. Assim, o algoritmo *Projection Pursuit* busca pela projeção que se apresenta menos gaussiana, com base no índice de projeção  $I(\alpha)$ .

O índice  $I(\alpha)$  é uma medida estatística univariada da não-gaussianidade da projeção, quanto maior o valor do índice, menos gaussiana é a projeção. Uma busca otimizada é usada para encontrar o vetor  $\alpha$  que maximiza o valor do índice, representando assim a projeção menos gaussiana da distribuição. Em seguida, os dados multivariados são transformados de maneira que a sua projeção ao longo do vetor  $\alpha$  seja uma distribuição gaussiana univariada. Sua fórmula de cálculo, assim como a busca pelo vetor  $\alpha$  que maximiza o valor desse índice estão descritos em detalhe em Friedman (1987), Barnett et al. (2012) e Barnett et al. (2014). A iteração repetida deste procedimento, buscando por projeções pouco gaussianas e as normalizando, transforma a base de dados original em uma distribuição multigaussiana para as K dimensões em estudo. Friedman (1987) também demonstra em detalhes o processo de gaussianização utilizado neste procedimento, sendo o mesmo um tipo de transformação *normal score* com alguns passos adicionais, uma vez que a transformação deve ser feita apenas sobre o vetor  $\alpha$ , deixando as demais direções ortogonais intactas (Barnett et al., 2014).

A Figura 3 mostra um exemplo de progressão dos dados à medida que se avançam as iterações desta etapa de *Projection Pursuit*, evidenciando o aumento da normalização dos dados e descorrelação.



**Figura 3: Progressão dos dados através do algoritmo *Projection Pursuit*. Fonte: Barnett *et al.* (2014). O gráfico superior mostra o índice de projeção  $I(\alpha)$  reduzindo à medida que avançam as iterações. Histogramas mostram a distribuição da projeção dos dados ao longo do vetor antes (à esquerda) e depois (à direita) da normalização em cada iteração.**

#### 2.4.4 Critério de Interrupção de Busca

Esta etapa foi introduzida por Barnett *et al.* (2014) e é feita através da

implementação de um algoritmo *bootstrapping*, em que uma simulação de Monte Carlo amostra aleatoriamente  $M$  distribuições das  $K$  variáveis, calculando o índice de projeção para todas as  $M \times K$  distribuições, gerando uma distribuição de índices de projeção referida por  $I$ . O usuário define um percentil desta distribuição  $I$  que deseja que o algoritmo de projeção busque como referência. Por exemplo, se o usuário escolher um percentil de 50%, então o *Projection Pursuit* irá proceder com as iterações até que o índice de projeção  $I(\alpha)$  seja menor que o percentil 50 da distribuição  $I$ , significando que os dados transformados são mais gaussianos do que 50% das amostras aleatórias. Caso o *target* não seja atingido dentro do número máximo de iterações estipulado, então o programa reportará o percentil atingido. A escolha de percentis altos (e.g., 99) significa transformação pouco gaussiana, enquanto a escolha de percentis baixos (e.g., 1) implica em transformação muito gaussiana.

#### 2.4.5 Retro-transformação

A retro-transformação é feita através do registro da tabela transformação obtida pelo mapeamento gaussiano (*Gaussian Mapping*), o qual mapeia a distribuição não-gaussiana  $K$ -dimensional em uma distribuição multi-gaussiana decorrelacionada, de dimensões correspondentes (Barnett *et al.*, 2014). Após a modelagem independente dos dados, a retro-transformação procede com o retorno dos nós simulados ao espaço real de acordo com os dados vizinhos (Figura 4).

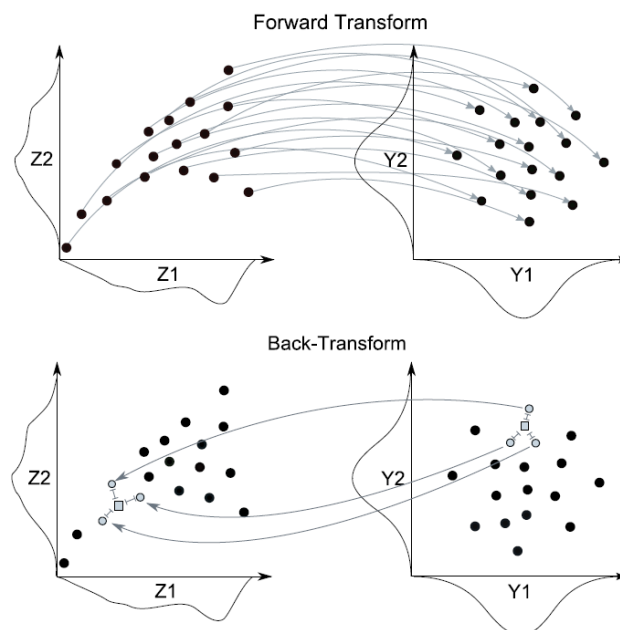


Figura 4: Esquema de mapeamento gaussiano. Fonte: Barnett *et al.* (2014).

Esta transformação deve ocorrer de tal forma que as posições espaciais relativas entre dados vizinhos devem ser mantidas no espaço real e transformado, minimizando mudanças nas distâncias entre os dados. A retro-transformação segue essa mesma lógica: nós simulados devem ser retro-transformados mantendo no espaço real as mesmas distâncias relativas aos dados circunvizinhos do espaço transformado.

## 2.5 CATEGORIZAÇÃO DE RECURSOS MINERAIS

Um dos objetivos deste trabalho é propor o uso do modelo simulado para mapeamento da incerteza espacial dos teores de múltiplas variáveis e, por fim, subdividir a área de estudo em setores com base na incerteza. O foco deste trabalho é categorizar os recursos minerais tendo em conta a alimentação da planta no curto prazo. Como a técnica aqui empregada se assemelha a um método de classificação de recursos minerais, neste capítulo serão abordados brevemente alguns métodos de categorização.

Os métodos de classificação de recursos minerais podem ser divididos em dois grandes grupos: os geométricos e os geoestatísticos. Os primeiros são os que levam em consideração, dentre outros fatores, a quantidade e localização dos dados disponíveis para estimativa. Já os métodos geoestatísticos, levam em consideração a continuidade e variabilidade espacial em intervalos de confiança definidos, obtidos principalmente por meio de simulação geoestatística. Revisões recentes a partir de declarações públicas de recursos minerais mostraram que grande parte (~75%) dos métodos aplicados na indústria são os geométricos, principalmente Espaçamento de Furos de Sondagem e Restrições de Vizinhança (Silva, 2015; Owusu & Dagdelen, 2019).

Segundo Souza *et al.* (2010), um método adequado de classificação de recursos deve oferecer a medida do nível de confiança das estimativas. Nesta linha de pensamento há a chamada regra dos 15%, a qual circula no meio da indústria mineral pelo menos desde a década de 1990 (Dohm, 2005). Esta regra empírica, afirma que o desvio associado a estimativa deve ser de até mais ou menos 15% com 90% de confiança, para um período de produção especificado. Geralmente, são abordados dois períodos de produção, um mensal e outro anual, nos quais esta regra é aplicada. Caso a variação esteja dentro do estabelecido pela regra dentro do período de

produção mensal, o recurso é chamado de medido, caso contrário, avalia-se o segundo período de produção; se a variação estiver dentro de 15% com 90% de confiança, então o recurso é classificado como indicado, do contrário, inferido. Os períodos de produção e percentuais aceitáveis de erro e nível de confiança podem variar de bem mineral para bem mineral e, também, entre depósitos, cabendo ao *Competent Person* (CP) tal julgamento.

Silva (2015) propôs dois métodos de classificação baseadas em técnicas geoestatísticas, sendo uma derivada da Variância de Krigagem, combinada com validação cruzada, e a outra com base na variabilidade de realizações simuladas dentro de painéis de produção e intervalos de confiança específicos, com base no desvio de 15% em relação à média para 95% das vezes, obtido por meio de janelas móveis. Em Silva (2020) tem-se um exemplo recente do uso da regra dos 15% aplicado à um depósito de bauxita, com uma análise da influência de diferentes malhas de sondagem na incerteza da classificação de recursos minerais. Saldanha (2020) demonstra o uso combinado entre os métodos geoestatísticos e geométricos, sendo o primeiro por meio de simulação condicional, com as incertezas medidas com base na regra dos 15%. A partir daí se define a melhor malha de sondagem para cada categoria. Assim, o resultado da classificação é relativamente simples e transparente ao investidor e, ao mesmo tempo, carrega consigo informações acerca da incerteza da classificação.

Abzalov & Bower (2014) apresentaram uma aplicação de técnica geoestatística para classificação de depósitos de bauxita. Os critérios propostos foram +/-10% de erro a um nível de confiança de 95%, considerando painéis de lavra com volumes trimestrais e anuais, para recursos medidos e indicados, respectivamente. Para recursos inferidos, o critério foi +/-30% a um nível de confiança de 95% considerando o volume total do depósito.

Nesta dissertação, o método que será apresentado segue o mesmo princípio da regra dos 15%. Porém, como o foco é a medição da incerteza no curto prazo, os períodos de produção considerados são de 2 e 4 semanas e o desvio aceitável estabelecido é de 5% com 90% de confiança. Além disso, neste trabalho também serão usados painéis de lavra variáveis definidos pelo método de clusterização de médias K, sendo uma forma de levar em consideração também possíveis mudanças

de polígonos de lavra pelo planejamento de mina. A próxima seção descreve os detalhes desta abordagem.

### 3 ESTUDO DE CASO

Neste capítulo será explicado, em maior detalhe, a metodologia adotada para este trabalho, sendo dividido em quatro partes principais. A primeira apresenta uma breve contextualização do problema e a justificativa para aplicação da técnica aqui estudada. A segunda apresenta a análise exploratória dos dados. A terceira parte está focada na aplicação da técnica PPMT para simulação geoestatística multivariada. E a última parte propõe o uso da técnica para medidas de incerteza dos teores em um depósito mineral e categorização dos recursos minerais, considerando a alimentação da planta metalúrgica no curto prazo.

#### 3.1 CONTEXTO

O trabalho foi desenvolvido em uma mina de níquel laterítico da mineradora Anglo American plc., localizada no município de Barro Alto - Goiás, onde as variáveis de teor do minério mais importantes para controle do processo metalúrgico são: Ni, Fe, SiO<sub>2</sub>, MgO e CaO. Além destas, existem também outras variáveis que impactam no processo (Guimarães, 2019), porém aqui será dado enfoque nas variáveis químicas mencionadas.

O teor de níquel para alimentação da planta é definido pela estratégia do planejamento de mina a fim de maximizar o valor presente líquido (NPV), geralmente tentando atingir teores acima de 1.5%.

Os teores de ferro devem ser limitados a um intervalo de aproximadamente 13% a 19%, para permitir um desempenho adequado dos fornos elétricos, principalmente em termos de consumo de energia e de qualidade na redução dos minerais da fase oxidada para metal livre. Valores elevados de ferro permitem um menor consumo de energia, mas exigem maior adição de agentes redutores para permitir a redução completa dos óxidos de ferro, e isso pode acarretar presença de carbono no produto final, fenômeno este chamado de carburação. Valores muito baixos de ferro implicam em maior consumo energético devido à baixa condutividade do banho (minério fundido), além de aumentar a viscosidade, o que dificulta o vazamento de escória e metal.

As outras três variáveis compreendem o que se chama de relação sílica-magnésio ( $SMR = SiO_2/[MgO+CaO]$ ), devendo esta permanecer próximo de 1.80. A relação entre estes elementos tem impacto direto na performance e manutenção dos



fornos, uma vez que altas relações implicam em consumo químico do material refratário constituinte das paredes dos fornos, maior viscosidade da fusão, que causa dificuldades operacionais no vazamento da escória, e maior consumo energético devido à maior resistividade. Por outro lado, uma baixa relação aumenta a fluidez, podendo causar riscos de segurança durante o vazamento da escória.

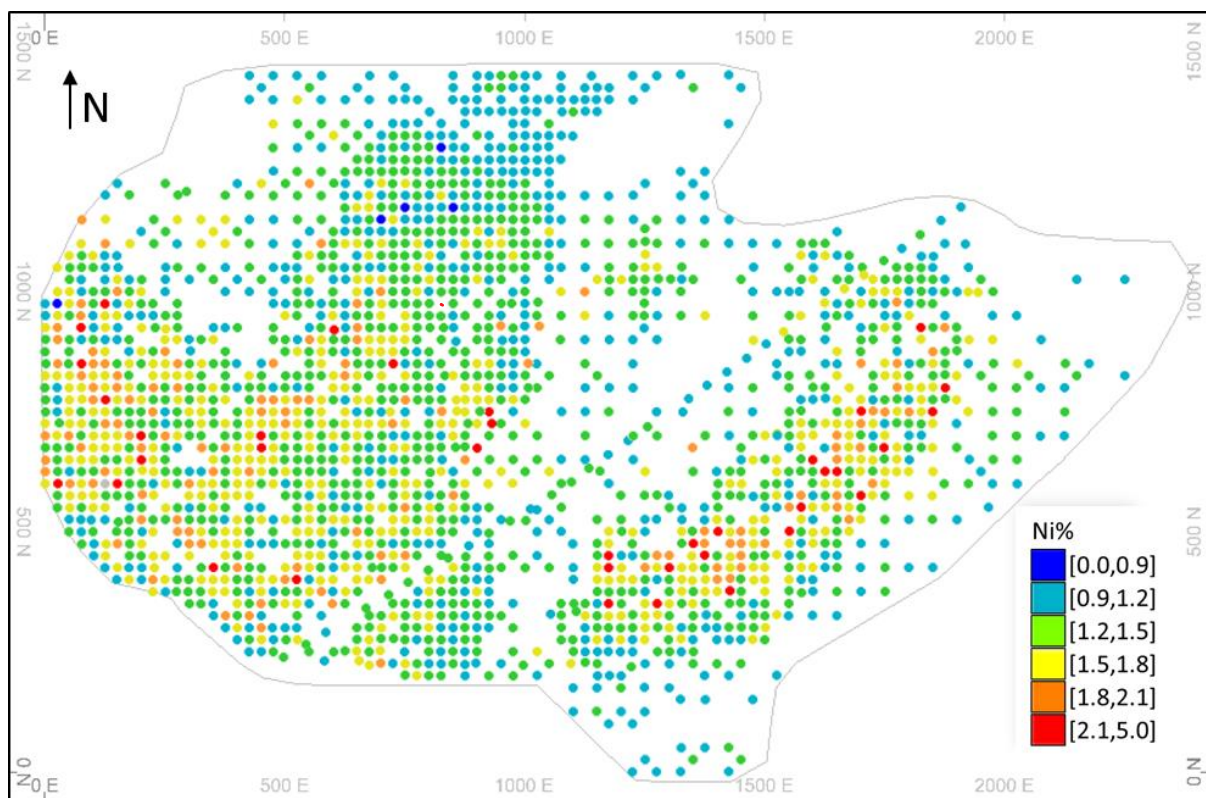
Neste contexto, é necessário que se estude e modele todas estas variáveis que atuam em conjunto no processo, levando a um melhor ou pior desempenho dependendo fortemente dos teores alimentados. Por esta razão, as correlações entre estas variáveis precisam ser bem representadas no modelo de teores, assim como o nível de confiança das estimativas associadas a elas.

### 3.2 ANÁLISE EXPLORATÓRIA DOS DADOS

A base de dados é composta por 6216 amostras de sondagem de circulação reversa (RC), provenientes de campanhas de perfuração para definição de recursos minerais. As amostras estão distribuídas por uma área de aproximadamente 2500 x 1500 m, em furos verticais de até 10 m de profundidade e média de espessura de minério em torno de 5 metros. O suporte amostral é de 1 metro e a malha de sondagem varia de 25x25m, nas partes mais adensadas, até aproximadamente 100x100m, nos locais onde a malha é mais espaçada. A Tabela 2 apresenta um sumário estatístico dos dados e a Figura 5 mostra um mapa com a distribuição espacial dos dados, com uma vista em planta e legenda por teores de níquel.

**Tabela 2: Sumário dos dados.**

	Ni	SiO <sub>2</sub>	MgO	Fe	CaO
Count	6216	6216	6210	6203	6205
Minimum	0.37	12.52	2.17	4.40	0.01
Maximum	4.55	53.15	38.20	38.30	2.41
Average	1.41	34.52	25.76	14.70	0.39
Variance	0.14	32.96	68.92	54.14	0.08

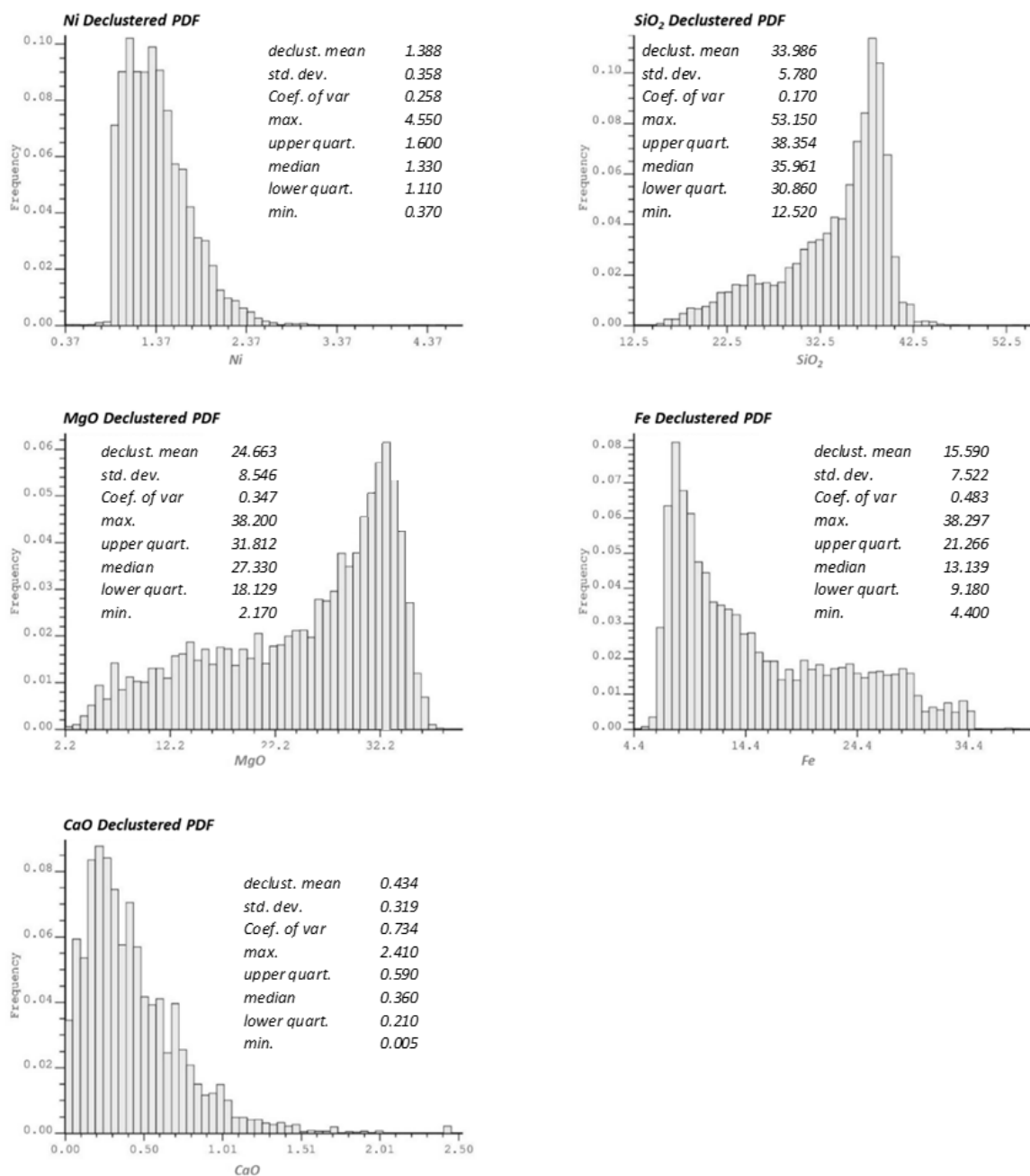


**Figura 5: Distribuição espacial dos dados. Vista em planta. Coordenadas locais.**

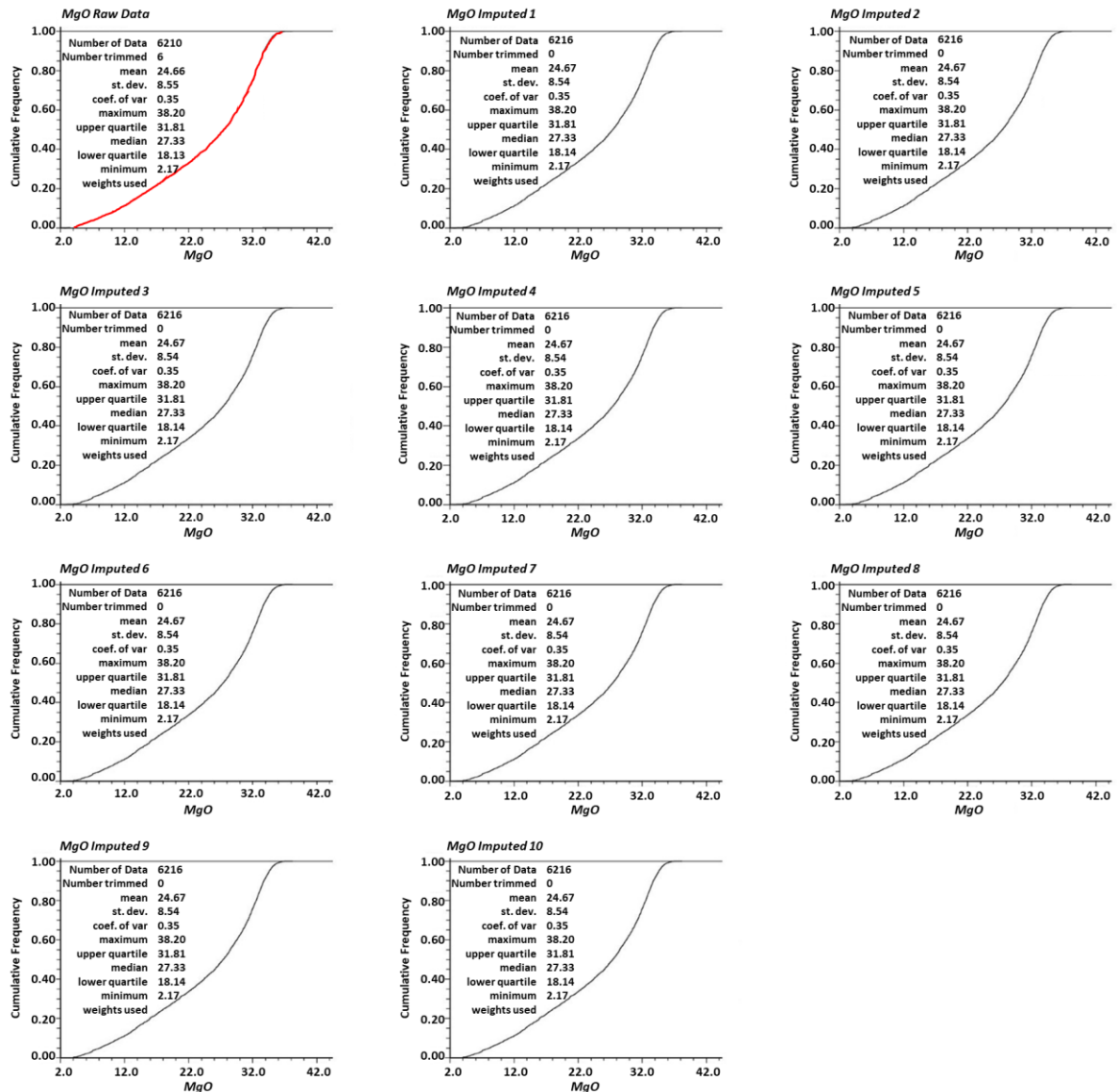
A Figura 6 exibe os histogramas dos dados usando pesos de desagrupamento para cada variável estudada. Os pesos de desagrupamento foram obtidos pelo método do vizinho mais próximo, o qual é uma aproximação do método de declusterização por polígonos (Isaaks & Srivastava, 1989) para casos tridimensionais.

Pelo sumário estatístico, nota-se que os dados são quase isotópicos, estando ausentes apenas algumas poucas observações de MgO, Fe e CaO. Como o método PPMT exige que a base de dados seja isotópica, estes dados ausentes foram preenchidos com uso da técnica de *data imputation* proposta por Barnett & Deutsch (2013). Para tal, primeiramente os dados foram transformados para o espaço gaussiano para então se proceder com o preenchimento dos dados faltantes. Foram geradas dez bases de dados isotópicas a partir da técnica de preenchimento e uma das bases de dados foi selecionada aleatoriamente para se prosseguir com a simulação. Esse procedimento é uma simplificação do fluxo completo proposto por Barnett & Deutsch (2013), no qual é sugerido o uso de uma realização de base de dados isotópica diferente para cada realização a ser simulada, de maneira a incorporar a incerteza do preenchimento da base de dados na simulação. No entanto, a simplificação adotada aqui foi considerada adequada visto que a proporção de

dados faltantes é muito baixa. A Figura 7 contém um exemplo do resultado do preenchimento de dados faltantes para a variável MgO, mostrando 10 histogramas acumulados (cdf) de bases de dados preenchidas, já retro-transformados para o espaço original, mais a base de dados original, para efeitos de comparação. Nota-se que praticamente não há variação entre as bases de dados preenchidas, pois o volume de resultados faltantes é muito baixo. O preenchimento dos dados para as variáveis Fe e CaO mostram resultados similares, com pequenas variações, e estão exibidos no Apêndice A.

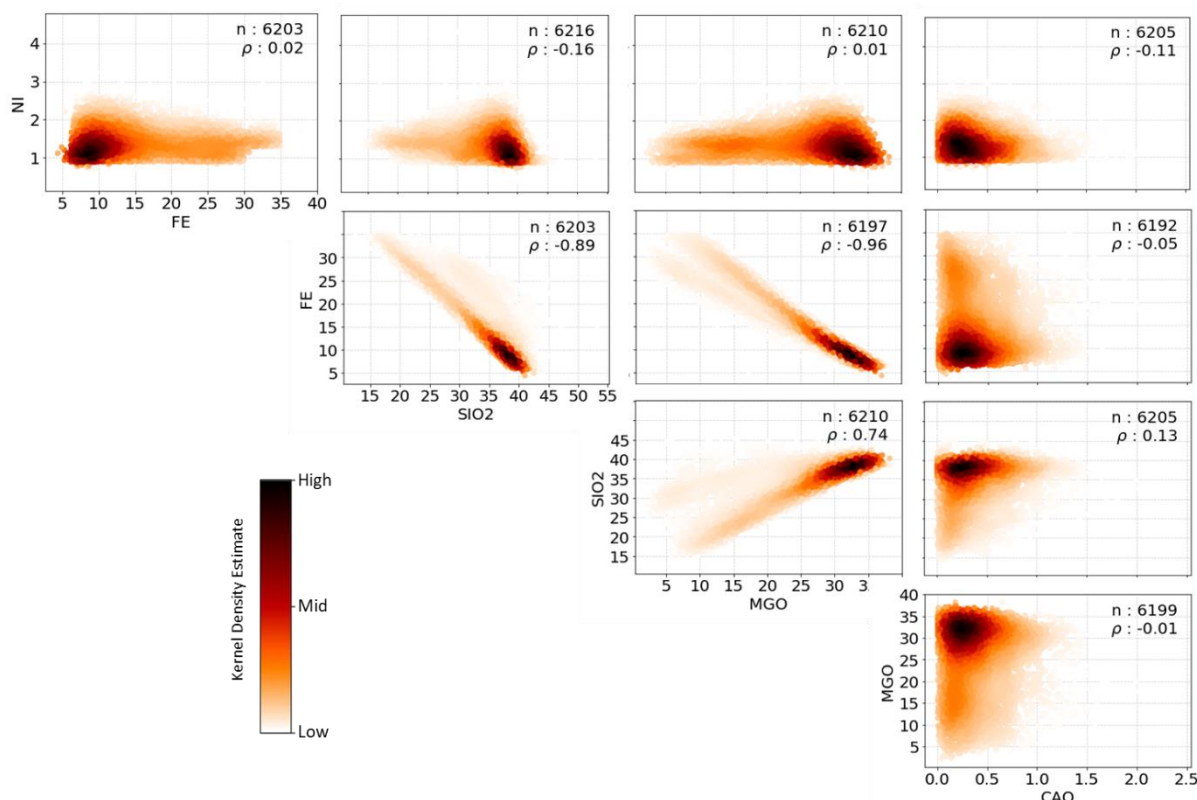


**Figura 6: Histogramas dos dados desagrupados para cada variável estudada.**



**Figura 7: Histogramas acumulados para MgO com dados isotópicos preenchidos (*Imputed* em preto) e comparação com dados originais (*Raw data* em vermelho).**

Os gráficos de dispersão entre as variáveis estão plotados na Figura 8. Nestes, é possível notar as relações complexas entre as variáveis, como heterocedasticidade e restrição, o que justifica o uso da técnica multivariada que permita a completa decorrelação entre as variáveis.



**Figura 8: Gráficos de dispersão entre as variáveis de estudo.**

### 3.3 TRANSFORMAÇÃO PPMT E SIMULAÇÃO

A transformação PPMT foi então realizada com uso do software ppmt.exe (CCG - *Center for Computational Geostatistics*, <http://www.ccgalberta.com/>), tendo como arquivo de entrada a base de dados isotópica criada a partir do preenchimento de dados e selecionada aleatoriamente. A transformação MAF também foi aplicada após a transformação PPMT, a fim de desconectar as distribuições para distâncias maiores que zero.

Foram executadas 100 iterações de *Projection Pursuit*, as quais foram suficientes para transformar os dados em distribuições multigaussianas e deixá-los completamente desconcorrelacionados. A Figura 9 mostra a evolução do índice de projeção à medida que se avança o número de iterações, mostrando que foram necessárias quase 100 iterações para se atingir um percentil de 50%, sendo este o índice alvo selecionado para esta transformação.

Na Figura 10, têm-se os histogramas e os gráficos de dispersão dos fatores PPMT no espaço transformado. Nota-se, que o fator correspondente a cada variável possui uma distribuição gaussiana padrão, com média zero e variância unitária, e são completamente desconcorrelacionados entre si ( $\rho = 0$ ). Ainda, a escala de cor indicando

a densidade de distribuição de pontos de maneira concêntrica indica multigaussianidade.

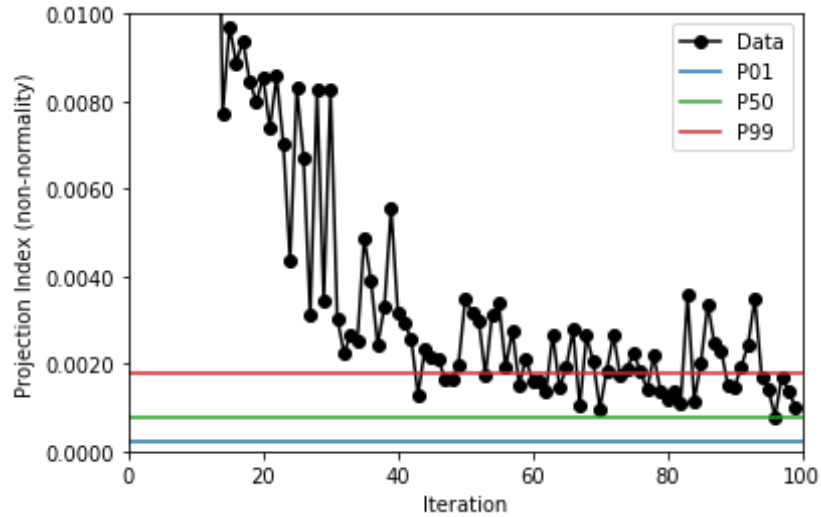


Figura 9: Índice de projeção (medida de não-normalidade) versus número de iterações.

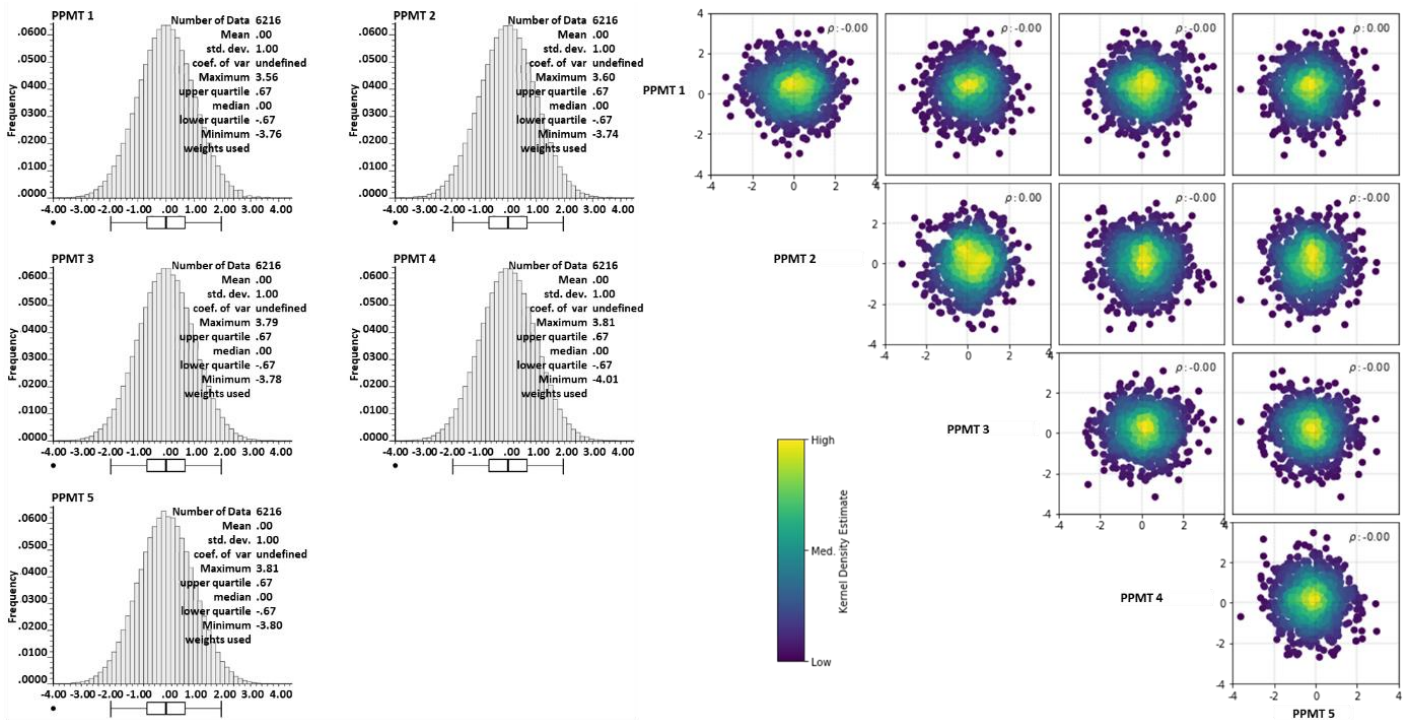
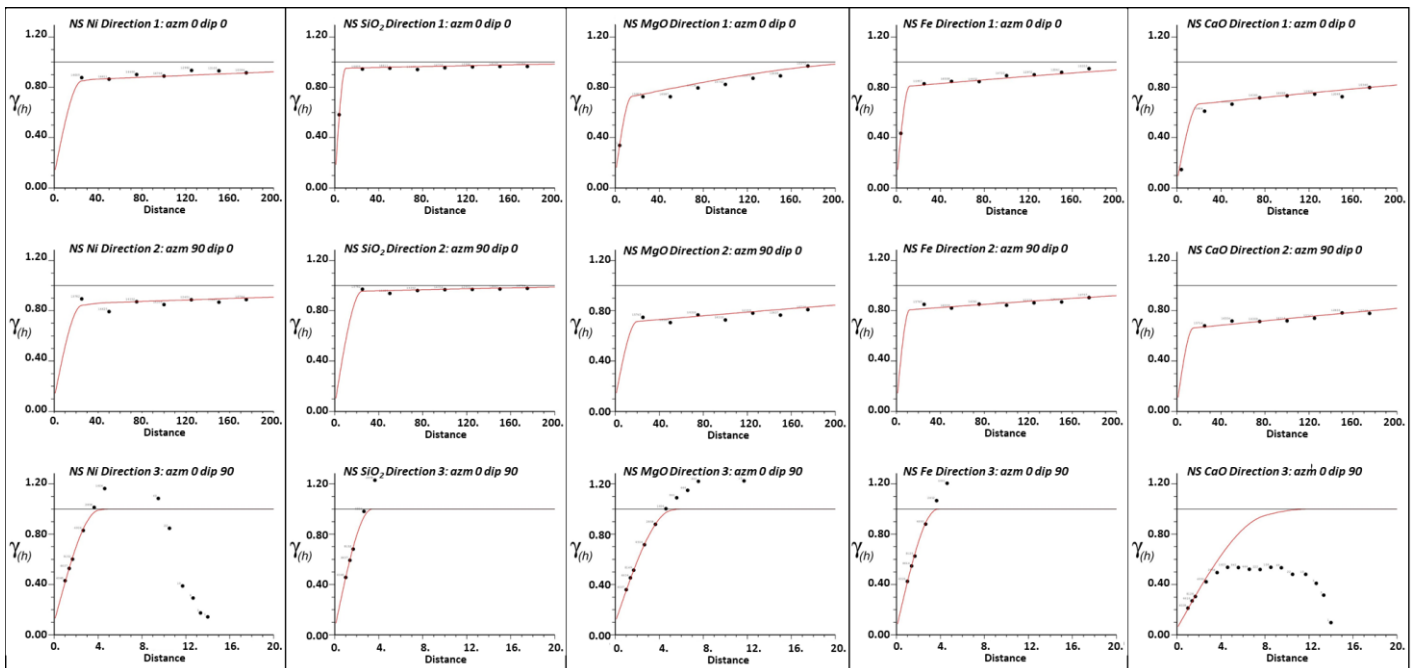


Figura 10: Histogramas e gráficos de dispersão dos fatores PPMT.

Barnett (2017) recomenda que os variogramas a serem utilizados na simulação sequencial gaussiana sejam modelados sobre as variáveis transformadas para o espaço gaussiano, e não diretamente sobre os fatores resultantes da transformação PPMT. A razão disso é que a remoção da dependência multivariada entre dados

regionalizados pode levar a uma desestruturação da sua continuidade espacial. Segundo o autor, o uso do modelo variográfico com base nos dados normais é a forma mais efetiva de reprodução dos variogramas originais dos dados, após a retro-transformação. De fato, isto foi verificado também neste trabalho. Foram feitos vários testes para calibração dos parâmetros da simulação, entre eles a variografia sobre os fatores PPMT e nscores (este último exibido na Figura 11). Estes demonstraram uma reprodução dos variogramas dos dados no espaço original ligeiramente melhores do que quando se utiliza os variogramas modelados sobre os fatores PPMT. Ainda assim, os variogramas dos nscores foram calibrados, tendo sua continuidade espacial inflada ou reduzida, para se tentar melhorar o resultado final das reproduções de distribuição e continuidade espacial dos dados no espaço real. Essa calibração é sugerida por Barnett (2015). Os testes para calibração dos parâmetros estão descritos em maior detalhe no Apêndice B.



**Figura 11: Variogramas direcionais dos nscores. Cada linha representa uma direção e cada coluna representa o nscore correspondente à uma variável.**

Com os fatores PPMT e variografia obtidos no passo anterior, a simulação sequencial gaussiana (SGS) foi executada com uso do programa usgsim.exe (CCG), gerando 50 realizações. Foi aplicado um elipsoide de busca de dimensões 150x150x30m, nas direções X, Y e Z, respectivamente. Estas direções coincidem com as principais direções de continuidade para a área de estudo. A malha de simulação foi de 1.25x1.25x1m, considerando o suporte amostral de 1m e o bloco de lavra (SMU)

de 5x5x2.5m. O número máximo de dados usados na simulação foi definido em 40.

Das figuras 12 a 18, estão exibidos os resultados e validações. Visualmente, as realizações mostram uma boa correspondência para o esperado para o depósito. A validação visual foi feita em planta (Figura 12) e em seções (Figura 13). A falta de correspondência exata entre dados e nós simulados se ao fato de alguns furos estarem um pouco descolados em relação à seção. De maneira geral, os histogramas das realizações reproduzem, de maneira satisfatória, o histograma dos dados desagrupados, apesar de algumas discrepâncias, as quais estão em um nível aceitável (Figura 14).

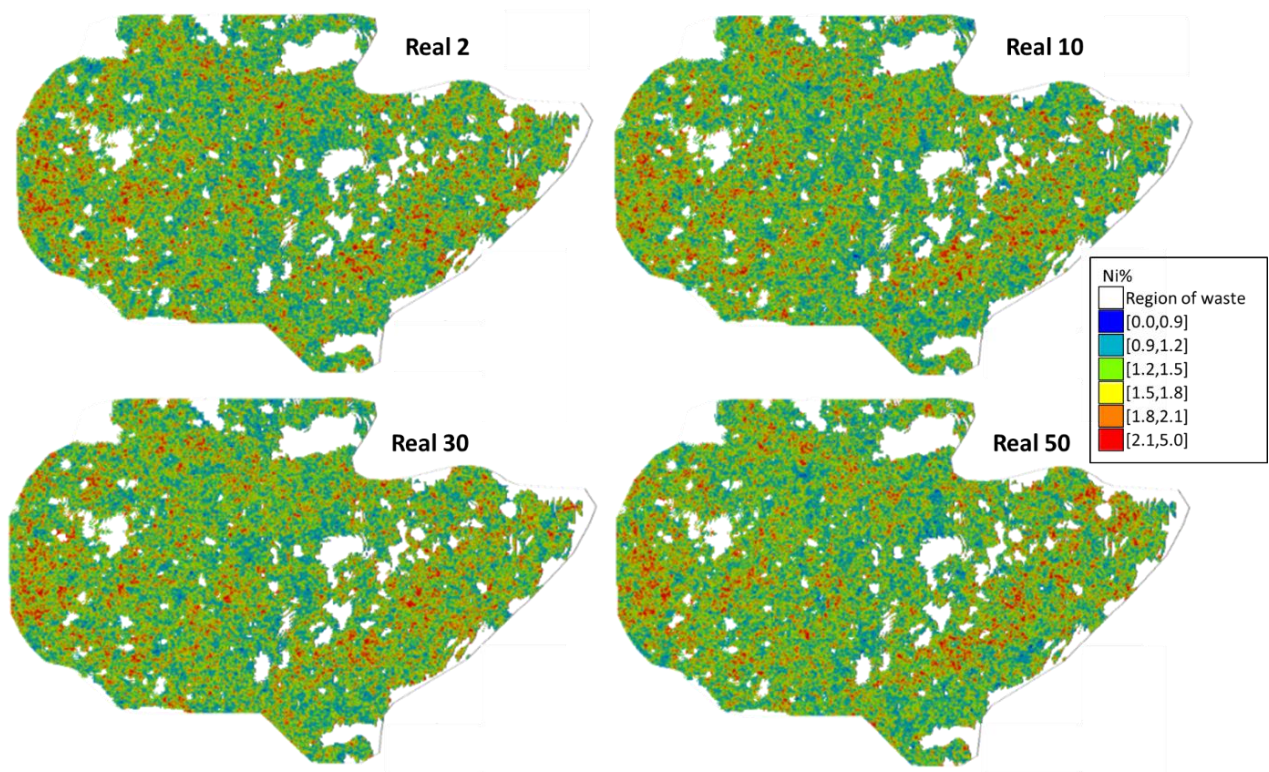
Já a reprodução dos variogramas, se mostrou um pouco mais complexa para ser obtida. Na validação exibida na Figura 15, os variogramas das realizações foram gerados através do programa *varsim.exe* (CCG), em que os nós simulados são tratados como *grid data*, ou seja, cada nó é comparado diretamente com outro nó em cada direção previamente determinada, não havendo tolerâncias angulares ou de largura de banda. Já os variogramas dos dados no espaço real foram obtidos indiretamente a partir da retro-transformação dos variogramas dos dados transformados e modelados no espaço gaussiano, por meio do programa *vmodelyz.exe* (CCG). De acordo com alguns autores (Vann & Sans, 1995; Wilde & Deutsch, 2005; Wilde & Deutsch, 2007; Deutsch & Kumara, 2020), esta técnica de inferência dos variogramas dos dados se aplica por apresentar variogramas mais estáveis e com menor influência de valores extremos, efeito proporcional e clusterização. Esta validação mostra uma boa reprodução dos variograms de Ni e MgO, um resultado intermediário para Fe e correspondências ruins para SiO<sub>2</sub> e CaO.

Considerando estes resultados, foram selecionadas seis realizações (1, 10, 20, 30, 40, 50) para ter sua continuidade espacial calculada como *point data*, usando os mesmos parâmetros de busca que foram usados para o variograma dos dados amostrais, permitindo também uma tolerância angular e na largura da banda de busca. Os resultados desta análise estão exibidos na Figura 16 e evidenciam algumas mudanças em relação à validação anterior. Fe e SiO<sub>2</sub> melhoraram a reprodução do variograma dos dados, enquanto Ni e MgO pioraram. A reprodução do variograma da variável CaO permaneceu com uma qualidade ruim, mas esta variável é considerada de menor importância comparada às outras, devido ao seu baixo teor, o que também pode afetar na qualidade da análise química. Além disso, de acordo com Barnett *et al.* (2014), a grande variedade de continuidade espacial entre as variáveis pode causar

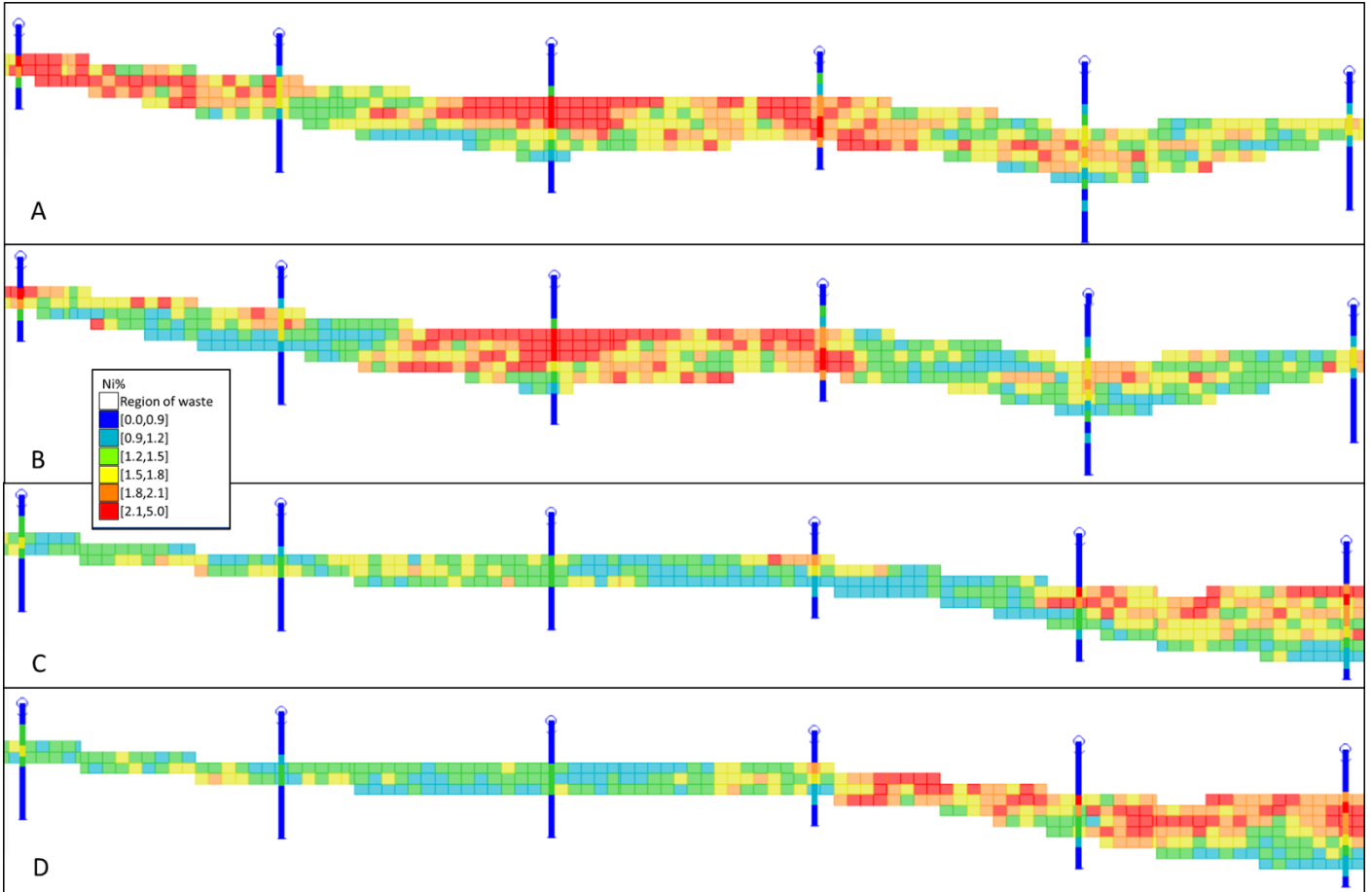


alguma dificuldade para recuperar a continuidade espacial única de cada variável, dependendo do grau de mistura que foi requerido para torná-las descorrelacionadas e multigaussianas nos cálculos dos fatores PPMT. Considerando este fato e a influência que a forma de se calcular a continuidade espacial tem na reprodução dos variogramas, os resultados foram aceitos e as simulações foram consideradas válidas para o propósito de cálculo da incerteza espacial e sua aplicação para categorização dos volumes planejados para lavra no curto prazo.

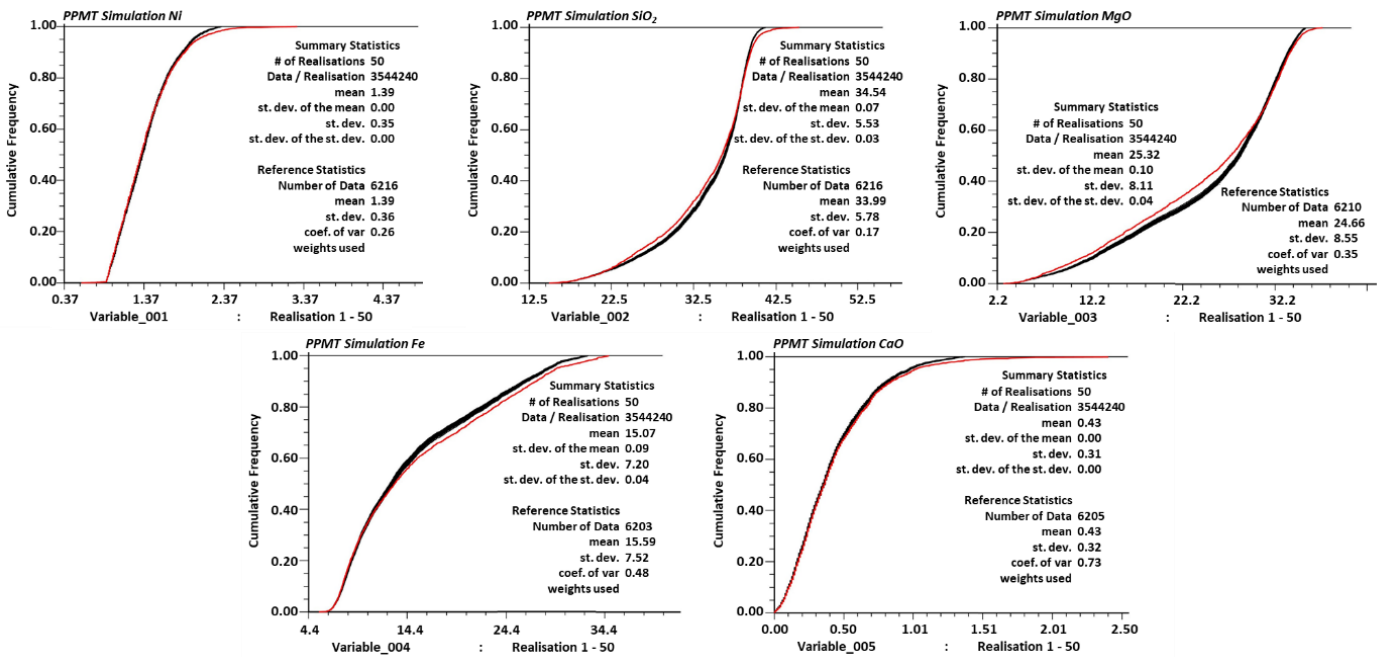
A Figura 17 mostra uma comparação das correlações entre as variáveis na base de dados com os nós de uma realização simulada e, também, com as estimativas resultantes da krigagem ordinária. Nota-se, claramente, uma melhora na reprodução das correlações da simulação em comparação com a krigagem. As relações complexas entre as variáveis também foram reproduzidas de maneira satisfatória, conforme demonstrado na Figura 18. Nota-se que, apesar da relativa grande dispersão de pontos, a maior concentração dos dados pareados segue o mesmo padrão do banco de dados amostral.



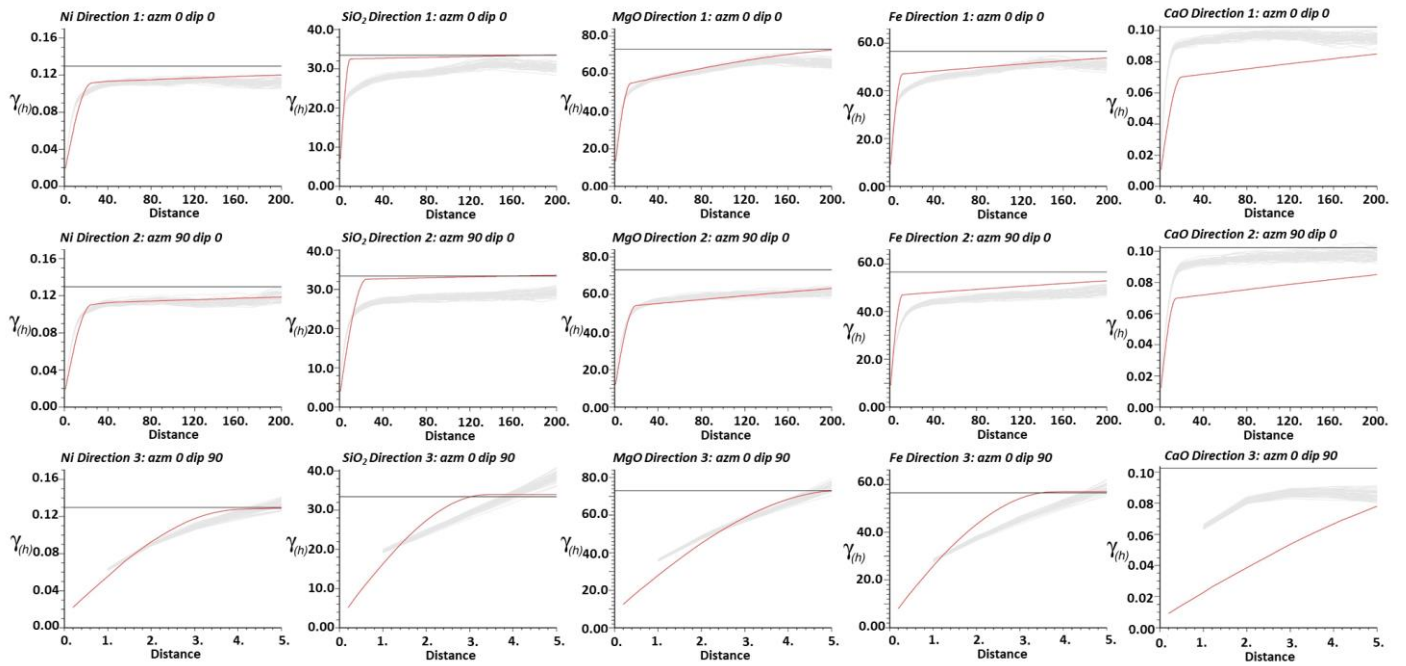
**Figura 12: Exemplo de quatro realizações simuladas para validação visual.**



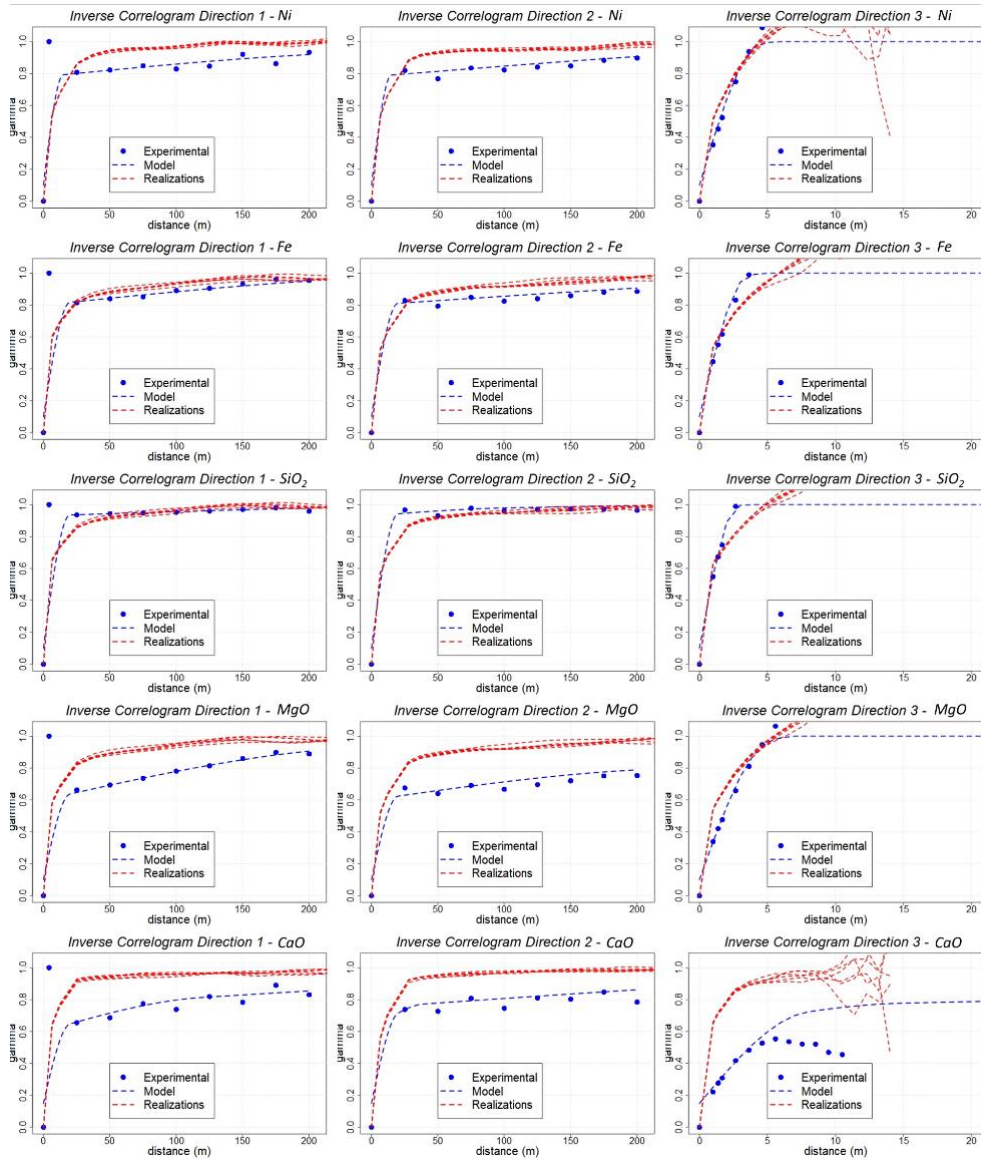
**Figura 13: Exemplo de seções para validação visual. A) seção 1-realização 1; B) seção 1-realização 2; C) seção 2-realização 1; D) seção 2-realização 2.**



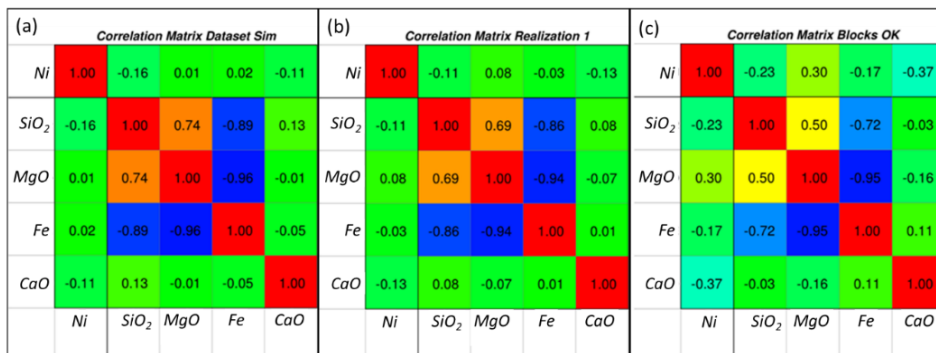
**Figura 14: Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto), para checagem da reprodução das distribuições de cada variável estudada.**



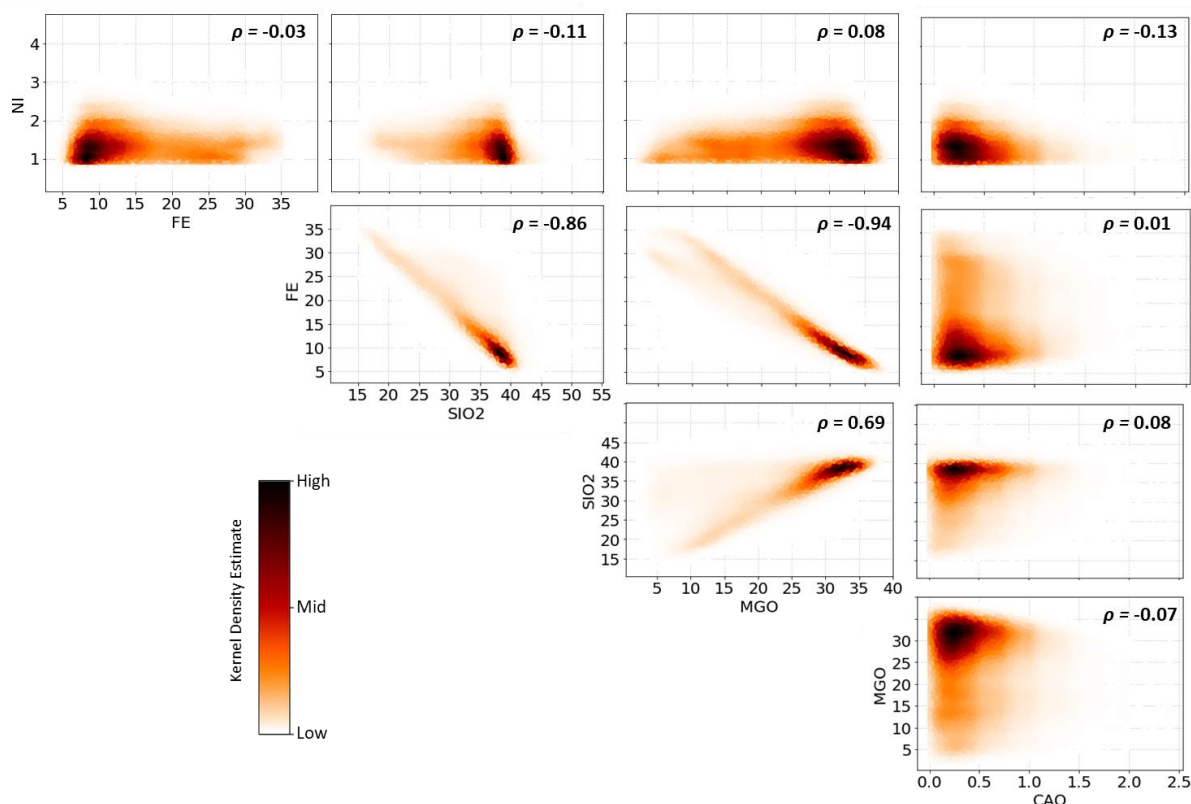
**Figura 15: Variogramas direcionais dos dados (vermelho) e das realizações simuladas (cinza), para checagem da reprodução da continuidade espacial (*grid data*). Cada linha representa uma direção e cada coluna representa uma variável de estudo.**



**Figura 16:** Variogramas direcionais dos dados (azul) e das realizações simuladas (vermelho), para checagem da reprodução da continuidade espacial (*point data*). Cada coluna representa uma direção e cada linha representa uma variável de estudo.



**Figura 17:** Matrizes para checagem da reprodução das correlações. (a) Dados originais; (b) Realização simulada; (c) Krigagem ordinária.

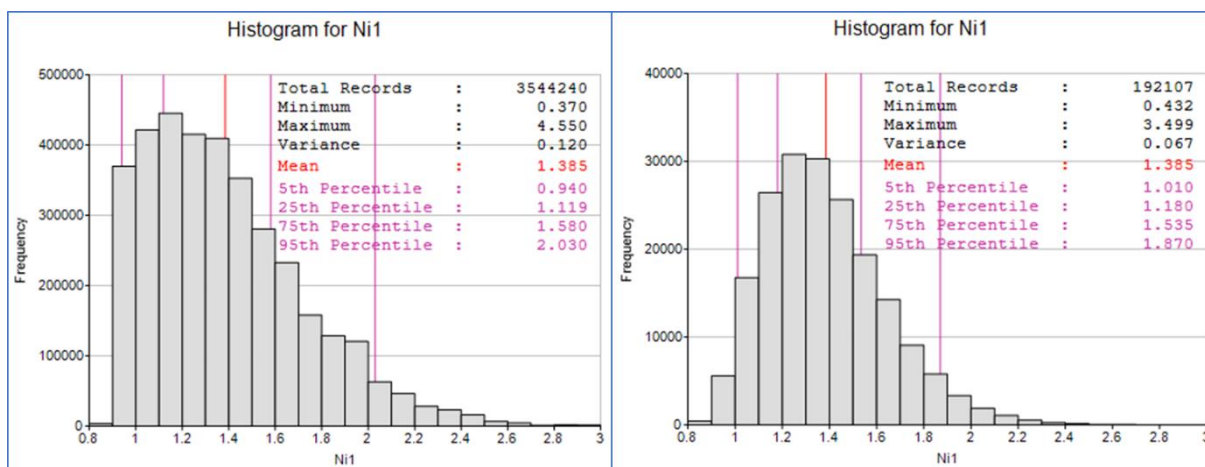


**Figura 18: Gráficos de dispersão entre as variáveis de estudo para a realização 1.**

### 3.4 ANÁLISE DE INCERTEZA ESPACIAL E CATEGORIZAÇÃO

A incerteza espacial foi mensurada por meio do coeficiente de variação (CV), o qual é uma medida estatística de dispersão relativa e standardizada (Marcisz, 2013).

Antes do cálculo da incerteza, as simulações foram convertidas a suporte de blocos de tamanho 5x5x2.5m (SMU) nas direções X, Y e Z, respectivamente, de maneira que os valores no suporte de blocos correspondam à média dos pontos simulados dentro do bloco. Essa regularização em suporte de blocos foi feita para que as medidas de variabilidade espacial pudessem ser tomadas considerando o tamanho da unidade mínima de lavra. Apesar de que a média dos teores dentro de um dado painel de lavra é a mesma considerando o suporte de pontos ou blocos, o coeficiente de variação é diferente. Além disso, apesar da lavra ocorrer em painéis, blocos individuais dentro de um painel podem ser enviados a diferentes destinos (minério de alto/médio/baixo teor e estéril), tornando a regularização para escala de SMU necessária. A Figura 19 ilustra o efeito da mudança de suporte para a variável Ni da realização 1. Como era esperado, a média se mantém constante e ocorre uma redução da variância.

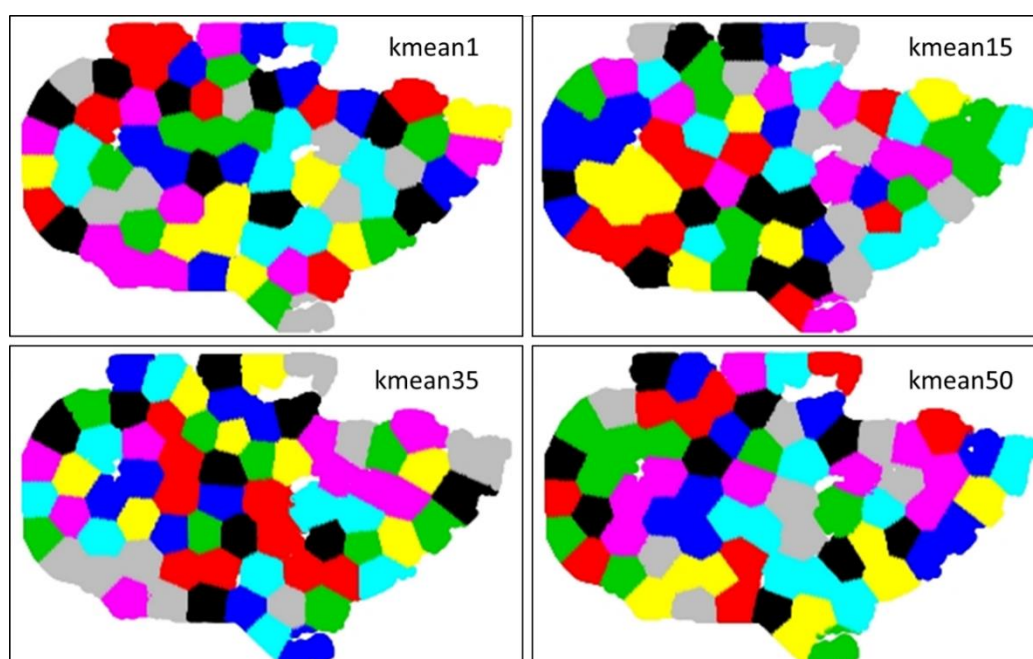


**Figura 19: Exemplo do efeito da mudança para a variável Ni da realização 1.**

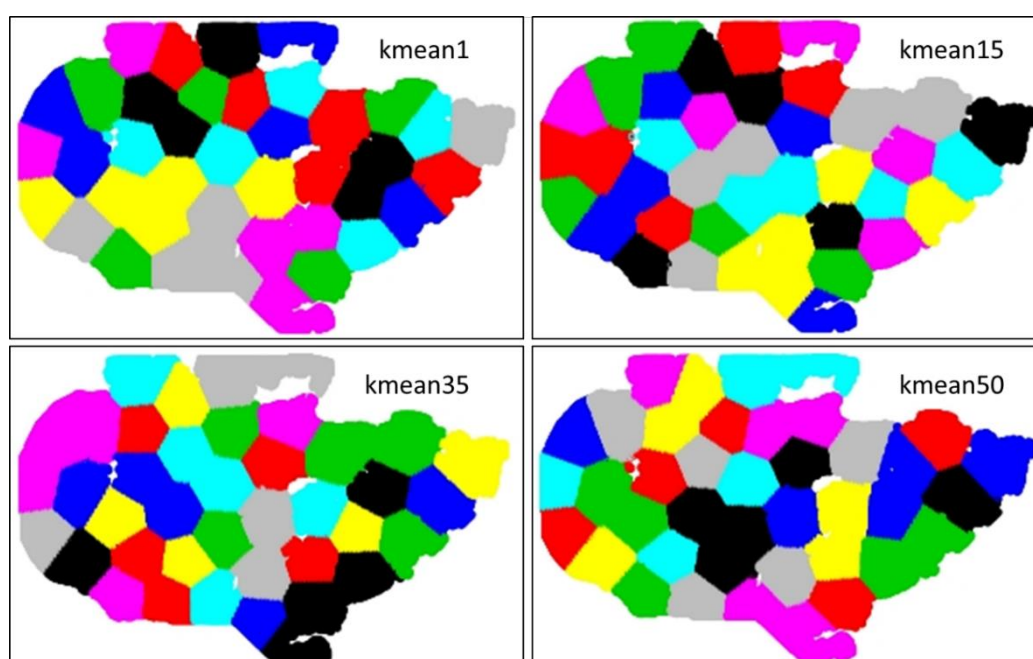
Para o cálculo do CV, foram gerados painéis representativos de lavra para volumes de produção correspondentes a períodos de duas e quatro semanas. O volume correspondente a duas semanas representa aproximadamente o equivalente a uma pilha de homogeneização (~75 kt) da etapa imediatamente antes da alimentação da planta industrial, a qual tem uma expectativa de variabilidade dos teores menor que 5% em torno do teor médio. Por esta razão, considerando esta perspectiva de curto prazo, foi assumido um CV de 5% como limite para definir as categorias de variabilidade dos painéis de lavra. O volume de produção correspondente a duas semanas permite a separação entre as categorias 1 e 2 (baixa e média variabilidade, respectivamente), enquanto o volume referente a quatro semanas permite a distinção entre as categorias 2 e 3 (média e alta variabilidade, respectivamente).

Para permitir menos variabilidade entre o número de blocos contidos em cada painel, os painéis foram definidos usando a técnica de clusterização por médias k (MacQueen, 1967) ao invés de se usar painéis retangulares. Também, para levar em consideração a possibilidade de se ter diferentes polígonos no momento da lavra, devido à mudança na estratégia de lavra ou mesmo por serem disponibilizadas novas informações geológicas que alterem o plano estabelecido, foram gerados 50 diferentes cenários de painéis com cluster definidos por médias k, mudando-se a semente inicial do algoritmo de clusterização, para ambos os períodos de produção. As figuras 20 e 21 exibem, respectivamente, exemplos de 4 diferentes cenários para os dois períodos de produção, de duas e quatro semanas. Nestas figuras, cada cor representa um cluster correspondente a um volume específico de produção.

Mesmo que o coeficiente de variação ( $CV=\sigma/\mu$ ) por si só não represente o nível de confiança de 90%, para representar este nível de confiança foi analisado o CV de todas as cinco variáveis dentro dos 50 cenários distintos de painéis de lavra, devendo o CV ser menor que 5% em 90% das vezes para o bloco ser classificado na categoria 1 (para o volume de duas semanas) ou categoria 2 (para o volume de quatro semanas). A categoria 3 foi atribuída aos blocos que não se enquadraram nas duas anteriores.



**Figura 20: Vista em planta de 4 diferentes cenários para o painel de duas semanas.**



**Figura 21: Vista em planta de 4 diferentes cenários para o painel de quatro semanas.**

O coeficiente de variação foi calculado para cada variável dentro de cada painel de lavra. O fluxo para se definir a categoria final foi:

- Selecione um polígono de lavra correspondente a duas semanas de produção;
- Calcule a média de cada realização simulada para a primeira variável dentro do polígono selecionado e, também, a média das médias (E-type);
- Calcule o CV para a primeira variável a partir da razão do desvio padrão das médias das realizações pelo E-type;
- Se o CV for menor que o limite de 5%, então todos os blocos dentro deste painel recebem um indicador igual a 1, caso contrário, o indicador é 0;
- Repita os passos anteriores para as outras variáveis;
- Repita os passos anteriores para todos os painéis de lavras;
- Repita o procedimento para todos os 50 cenários de painéis de lavra;
- Avalie o número de vezes que cada bloco recebeu o indicador 1. Se a proporção de vezes for maior que 90%, então o bloco será classificado na categoria 1 (baixa variabilidade), do contrário recebe a categoria 2 (média variabilidade);
- Repita todo o procedimento anterior para o período correspondente a quatro semanas de produção, mas agora mudando o resultado entre as categorias 2 e 3 (média e alta variabilidade, respectivamente).

Por este procedimento, o número máximo de vezes que o bloco pode receber o indicador 1 é dado pelo número de cenários diferentes de lavra, que no caso são 50, multiplicado pelo número de variáveis. Uma vez que o plano de lavra pode mudar devido a várias razões, a simulação de 50 cenários distintos de lavra por médias  $k$  permite que esta análise de incerteza leve em consideração tais variações, ainda que de maneira virtual. Isto porque, um determinado bloco pode receber o indicador 1 por estar em um painel com coeficiente de variação menor que 5%, mas poderia receber um indicador 0 caso a configuração do polígono de lavra fosse diferente, mudando sua categoria. Assim, para cumprir com o critério aqui proposto, o bloco deve ter o CV menor que 5% em pelo menos 90% dos diferentes cenários e para todas as variáveis em conjunto.

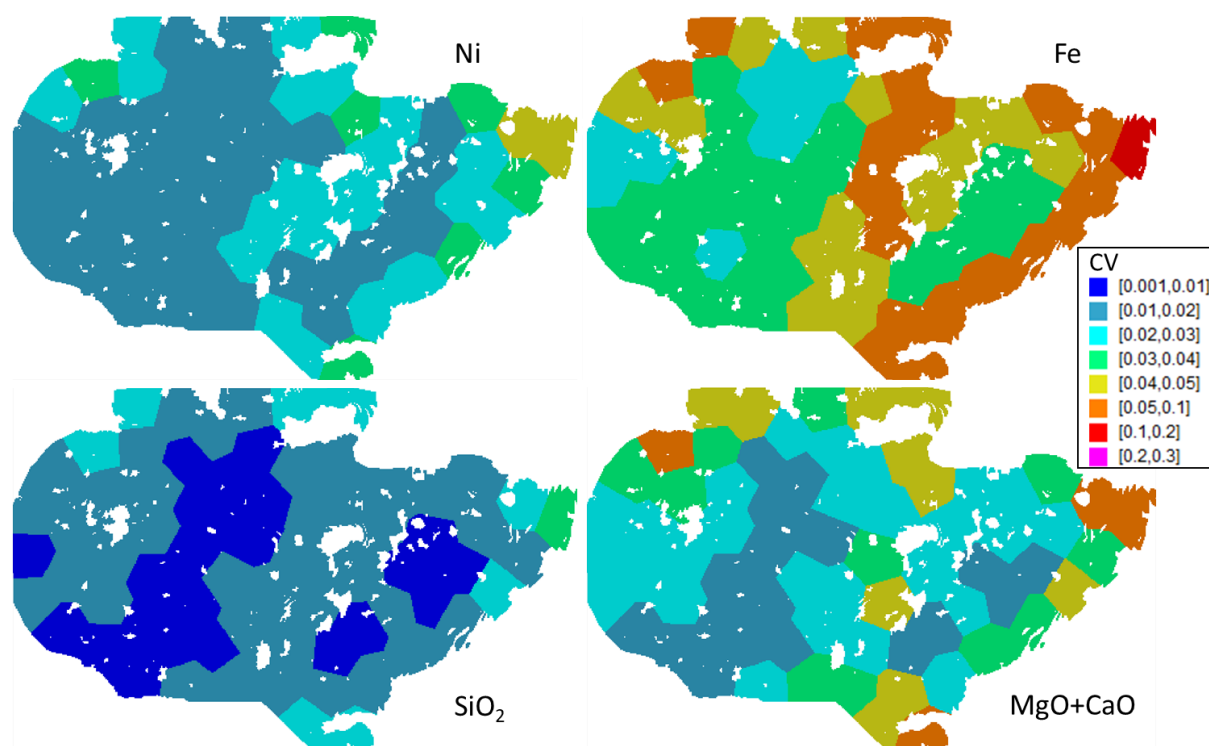
Claro que esta abordagem representa um critério bastante rigoroso, mas para o propósito deste trabalho, medir a incerteza e definir classes de



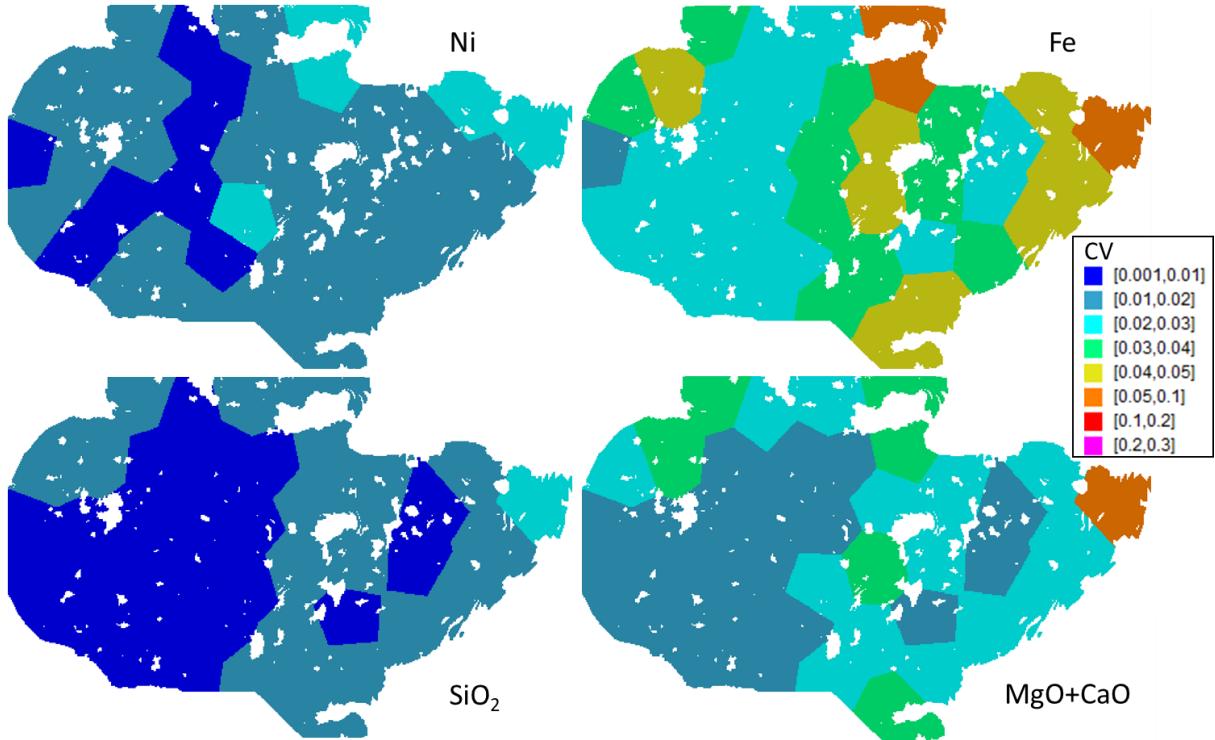
confiabilidade/variabilidade considerando o planejamento de curto prazo, a proposta foi considerada adequada. De qualquer modo, os critérios podem ser ajustados para diferentes propósitos e depósitos minerais, usando períodos de produção maiores, assim como coeficientes de variação maiores.

As figuras 22 e 23 exibem os mapas com os coeficientes de variação obtidos para as variáveis estudadas em um dos cenários de lavra simulados pela técnica das médias  $k$ , considerando os volumes de produção de duas e quatro semanas, respectivamente. Nestas figuras é possível ter uma ideia de quais locais e variáveis apresentam maior variabilidade e que, por isso, poderiam representar um maior problema para a alimentação da planta. Por questões práticas, a variável MgO foi somada à variável CaO e está apresentada como uma única variável: MgO+CaO.

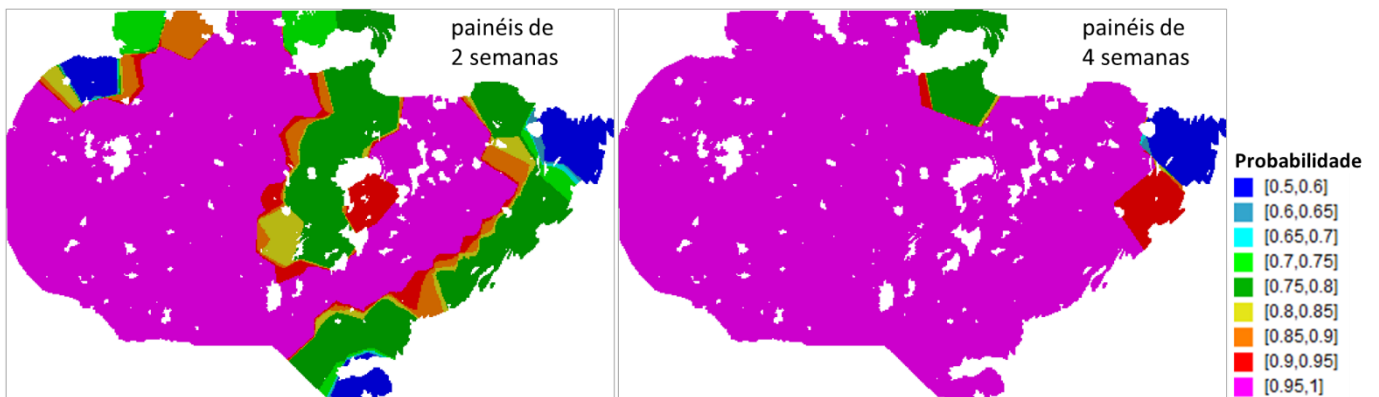
Após a repetição do procedimento de cálculo do CV para todos os cenários, é possível avaliar a distribuição espacial das probabilidades de cada bloco ter um CV menor que 5%, levando em consideração todas as variáveis estudadas e todos os cenários de lavra simulados (Figura 24). Com este resultado é possível prosseguir para a categorização final, tendo como critério CV menor que 5% com pelo menos 90% de probabilidade.



**Figura 22: Mapa dos coeficientes de variação calculados para as variáveis estudadas, considerando volume de produção correspondente a 2 semanas para um cenário de lavra.**

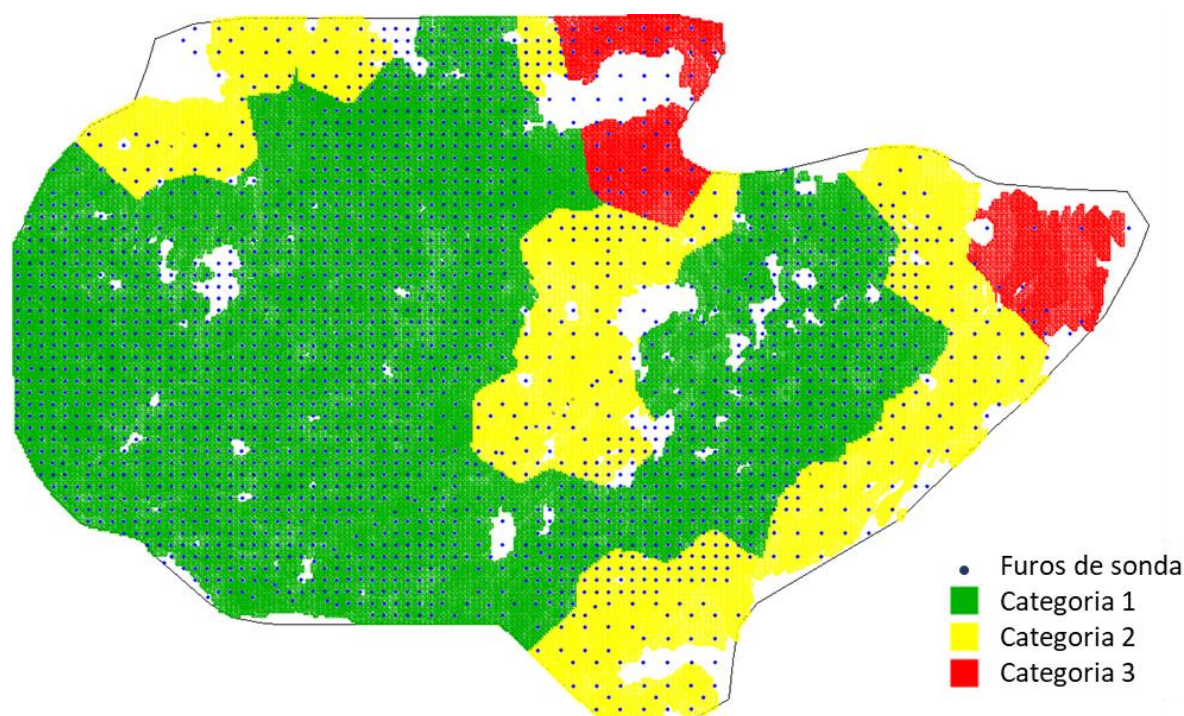


**Figura 23:** Mapa dos coeficientes de variação calculados para as variáveis estudadas, considerando volume de produção correspondente a 4 semanas para um cenário de lavra.



**Figura 24:** Mapa de probabilidades de o CV ser menor que 5%, considerando todas as variáveis de maneira conjunta e em todos os cenários de lavra simulados pelas médias k.

A Figura 25 mostra o resultado final, com a área de estudo dividida nas três categorias de variabilidade para o horizonte de curto prazo. É possível observar que as categorias estão condizentes com a distribuição dos dados. Locais com mais informação (pontos azuis) entraram na categoria 1 (baixa variabilidade entre as realizações), seguidos pela categoria 2 onde os dados estão um pouco mais espaçados e, finalmente, categoria 3 (alta variabilidade entre as realizações) nas áreas com menos informação ou alta variabilidade natural.



**Figura 25: Mapa com a vista em planta de distribuição das categorias.**

Ressalta-se aqui que as realizações simuladas foram regularizadas em blocos maiores para refletir a unidade seletiva de lavra (SMU), cujo tamanho é 5x5x2.5 m.

A principal vantagem dessa técnica é que o uso da simulação agrega uma medida de incerteza espacial associada aos valores dos blocos de lavra. Isso permite que o planejamento de lavra seja feito considerando a expectativa de variabilidade do processo metalúrgico. Regiões enquadradas na categoria 1, por exemplo, teriam uma expectativa de variabilidade conjunta para todas as variáveis de até 5% (CV) para pelo menos 90% das vezes. Além disso, essa informação também pode ser usada para guiar planos de sondagem, priorizando obtenção de novas informações nas porções mais ricas do depósito, as quais geralmente entram em produção no curto/médio prazo, mas que se enquadrem nas categorias 2 e 3, as quais possuem média e alta variabilidade, respectivamente.

#### 4 CONSIDERAÇÕES FINAIS

Este capítulo apresenta as considerações finais acerca do trabalho aqui apresentado, com as devidas conclusões e recomendações para trabalhos futuros.

Este trabalho primeiramente apresentou uma revisão bibliográfica sobre os métodos tradicionais aplicados na indústria mineral, como a krigagem ordinária, a qual é amplamente utilizada, assim como outros métodos de geoestatística multivariada. Estes métodos não atingem por completo o objetivo aqui proposto, seja devido à própria construção da técnica, por vezes limitada a descobertas de correlações lineares, ou pela dificuldade de implementação prática devido à complexidade de modelamento dos chamados Modelos Lineares de Corregionalização (MLC) para casos com número de variáveis maior do que três. Foi também feita a revisão bibliográfica da técnica proposta por Barnett *et al.* (2014), *Projection Pursuit Multivariate Transform* (PPMT), a qual se propõe a endereçar o problema aqui apresentado, proporcionando a geração de realizações simuladas para casos multivariados de correlações complexas de forma a manter as correlações iniciais dos dados nos resultados da simulação.

O método PPMT foi então aplicado em um depósito mineral de níquel laterítico, cujas principais variáveis químicas para o processo metalúrgico são: Ni, Fe, SiO<sub>2</sub>, MgO e CaO. Antes da aplicação deste método de fatorização, foi feito o preenchimento da base de dados para torná-la isotópica, seguida então da transformação PPMT e posterior simulação sequencial gaussiana com 50 realizações independentes para cada variável. Por último, a retro-transformação do espaço PPMT para o espaço original.

As validações das reproduções dos histogramas, variogramas e correlações dos dados se mostraram aceitáveis, apesar de pequenas diferenças. No caso da reprodução da continuidade espacial, destaca-se a influência da forma de cálculo dos mesmos na reprodução dos variogramas dos dados, podendo apresentar resultados melhores ou piores. Os resultados foram considerados aceitos para se prosseguir para a próxima etapa de análise das incertezas espaciais.

A incerteza espacial foi calculada por meio do coeficiente de variação (CV) das realizações dentro de painéis de lavra. Estes foram obtidos com a técnica de clusterização por médias k para períodos de duas e quatro semanas. Para levar em consideração possibilidades de mudanças nos polígonos de lavra, foram gerados 50

cenários diferentes de painéis de lavra para ambos os períodos considerados. O CV foi então calculado considerando todas as variáveis dentro de cada painel e para todos os cenários. O nível de confiança estabelecido de 90% foi atingido quando o bloco a ser classificado recebeu indicador 1 ( $CV < 5\%$ ) para pelo menos 225 vezes das 250 vezes possíveis (número de variáveis vezes o número de cenários).

A subdivisão final nas categorias 1, 2 e 3 mostrou boa aderência ao esperado, com blocos de lavra na categoria 1 nas áreas de maior adensamento de dados, passando pela categoria 2 e, finalmente, categoria 3; esta última localizada nas bordas, geralmente associada a pouca quantidade de informação e/ou alta variabilidade natural.

#### 4.1 CONCLUSÕES

Tendo em consideração que a meta desta dissertação foi investigar uma alternativa para estimativa em depósitos multivariados com correlações complexas, com dois objetivos específicos associados: avaliar a aplicabilidade do método PPMT e investigar o mapeamento da incerteza dos teores, conclui-se que os objetivos desta dissertação foram atingidos, destacando-se aqui dois pontos de vantagem do método aqui estudado em relação ao método atualmente utilizado no depósito (krigagem ordinária):

1. A simulação com uso de PPMT permite a reprodução das correlações nas realizações simuladas, acarretando uma maior previsibilidade conjunta para planos de lavra e de alimentação da planta industrial;
2. A análise de múltiplas realizações possibilita um mapeamento de incerteza dos teores baseado numa medida de variabilidade espacial, com nível de variabilidade e confiança definidos.

Com estes resultados, a planta metalúrgica tende a uma maior estabilidade operacional, pois tem sua alimentação baseada em um plano de lavra construído sobre uma expectativa de variabilidade conjunta dos teores no curto prazo. O resultado também contribui para priorização dos planos de sondagem em áreas de maior variabilidade.

## 4.2 RECOMENDAÇÕES

Conforme observado nas validações, houve uma certa diferença nas reproduções dos histogramas e variogramas. Os histogramas das realizações apresentaram muito baixa variação, de forma que em alguns trechos os histogramas das realizações não continham o histograma dos dados. Alguns fatores que podem contribuir para o aumento das flutuações ergódicas envolvem o uso de uma semente diferente para cada realização e o uso de uma base de dados isotópica (proveniente do procedimento de *data imputation*) diferente para cada realização. Apesar dessas possibilidades de melhoria, importante levar em consideração que as amostras não representam necessariamente com exatidão a população em si, devido a fatores como erros amostrais e número de dados disponíveis; de forma que a busca pela reprodução a qualquer custo pode representar uma distorção de uma possível realidade representada pela simulação. Por isso a análise crítica é importante para avaliar se o intervalo de variação obtido, levando em consideração as realizações simuladas mais os dados amostrais, são considerados aceitáveis dentro do universo de possibilidades esperado para o depósito em estudo.

Já com relação aos variogramas, a melhoria no resultado de reprodução deles se mostra um tanto mais complexa, principalmente levando em consideração a forma de cálculo dos mesmos. Outros testes de calibração das continuidades dos variogramas no espaço transformado podem ser investigados, gerando variogramas com continuidade espacial inflada ou reduzida, combinado com alterações nos parâmetros de busca, incluindo número de amostras e raios de busca, podendo-se aplicar parâmetros específicos para cada variável. Além disso, sugere-se também uma análise variográfica tridimensional, para refinamento das principais direções de continuidade, e uso de variogramas gerados com pesos de declusterização (Machuca-Mory & Deutsch, 2007; Machuca-Mory & Deutsch, 2013).

Outra validação que seria interessante inserir no fluxo de trabalho é o teste de acuracidade e precisão (Deutsch, 1997). Esta validação permitiria verificar o quão o modelo simulado está acurado e preciso em relação aos dados amostrados.

Sugere-se também a avaliação futura do uso da técnica aqui proposta para a classificação de recursos minerais com foco no longo prazo, para fins de declarações públicas de recursos minerais. Neste caso, os períodos de produção e coeficiente de variação aceitável devem ser ajustados para refletir as necessidades e expectativas

de variabilidade no longo prazo, cabendo ao *Competent Person* (CP) tal julgamento.

## REFERÊNCIAS

- ABZALOV, M. Z.; BOWER, J. Geology of bauxite deposits and their resource estimation practices. *Applied Earth Science*, v. 123, n. 2, p. 118-134, 2014.
- BARNETT, R. M. *Managing Complex Multivariate Relations in the Presence of Incomplete Spatial Data*. Thesis (Doctor of Philosophy in Mining Engineering). University of Alberta, Edmonton, 2015.
- BARNETT, R. M. *Projection Pursuit Multivariate Transform*. Edmonton, 2017. Geostatistics Lessons. Disponível em: <http://geostatisticslessons.com/lessons/ppmt>. Acesso em: 17 jun. 2020.
- BARNETT, R. M.; DEUTSCH, C. V. Imputation of Geologic Data. *CCG Annual Report*, v. 15, n. 102, p. 1-16, 2013.
- BARNETT, R. M.; MANCHUK, J. G.; DEUTSCH, C.V. Projection Pursuit Multivariate Transform. *CCG Annual Report*, v. 14, n. 103, p. 1-18, 2012.
- BARNETT, R. M.; MANCHUK, J. G.; DEUTSCH, C. V. Projection Pursuit Multivariate Transform. *Mathematical Geosciences*, v. 46, n. 3, p. 337-359, 2014.
- BASSANI, M. A. A. *Uso de Dados de Diferente Suporte em Geoestatística e Desenvolvimentos em Simulação Geoestatística Multivariada*. Tese (Doutorado em Engenharia), Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2018.
- BASSANI, M. A. A.; COSTA J. F. C. L.; DEUTSCH C. V. Multivariate geostatistical simulation with sum and fraction constraints. *Applied Earth Science*, v. 127, n. 3, p. 83-93, 2018.
- BATTALGAZY N.; MADANI N. Categorization of mineral resources based on different geostatistical simulation algorithms: a case study from an iron Ore deposit. *Natural Resources Research*, v. 28, n. 4, p. 1329-1351, 2019.
- BOEZIO, M. N. M. *Estudos das Metodologias Alternativas da Geoestatística Multivariada Aplicadas a Estimativa de Teores de Depósitos de Ferro*. Tese (Doutorado em Engenharia), Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2010.
- CLARK, I. *Practical Geostatistics*. Alcoa: Geostokos Limited, 2001.
- DEUTSCH, C. V. Direct assessment of local accuracy and precision. In: BAAFI, E. Y.; SCHOFIELD, N. A. (eds.). *Geostatistics Wollongong '96*. Dordrecht: Kluwer Academic Publishers, 1997, p. 115-125.
- DEUTSCH, C. V.; JOURNEL, A. G. *GSLIB: Geostatistical Software Library and User's Guide*. New York: Oxford University Press, 1998.



DEUTSCH, C. V., KUMARA, P. *Transforming a Variogram of Normal Scores to Original Units*. Edmonton, 2017. Geostatistics Lessons. Disponível em: <http://www.geostatisticslessons.com/lessons/convertnsvariograms>. Acesso em: 24 jun. 2020.

DOHM, C. Quantifiable Mineral Resource Classification: A Logical Approach. In: LEUANGTHONG, O.; DEUTSCH, C.V. (eds.). *Geostatistics Banff 2004*. Quantitative Geology and Geostatistics, v. 14. Dordrecht: Springer, 2005. p. 333-342.

MARCISZ, M. Practical Application of Coefficient of Variation. In: SÁNCHEZ, A.; KRZEMIEN, A.; FERNÁNDEZ, P. R.; RODRÍGUEZ, F. J. I. (eds.). *Proceedings of the 13th International Congress on Energy and Mineral Resources (CIERM 2013)*. Santander: National Association of Spanish Mining Engineers, 2013. p. 202-208.

FRIEDMAN, J. H. Exploratory Projection Pursuit. *Journal of the American Statistical Association*, v. 82, n. 397, p. 249-266, 1987.

FRIEDMAN, J. H.; TUKEY, J. W. A Projection Pursuit Algorithm for Exploratory Data Analysis. *IEEE Transactions on Computers*, v. C-23, n. 9, p. 881-890, 1974.

GOOVAERTS, P. *Geostatistics for Natural Resources Evaluation*. New York: Oxford University Press, 1997.

GUIMARÃES, C. G. G. *Caracterização Tecnológica de Minério de Níquel Laterítico*. Dissertação (Mestrado em Engenharia), Curso de Pós-Graduação em Engenharia Metalúrgica, Materiais e de Minas. Universidade Federal de Minas Gerais, Belo Horizonte, 2019.

HUBER, P. J. Projection Pursuit. *The Annals of Statistics*, v. 13, n. 2, p. 435-475, 1985.

ISAAKS, E. H. *The Application of Monte Carlo Methods to the Analysis of Spatially Correlated data*. Thesis (Doctor of Philosophy), Department of Applied Earth Sciences. Stanford University, Stanford, 1990.

ISAAKS, E. H.; SRIVASTAVA, M. R. *An Introduction to Applied Geostatistics*. New York: Oxford University Press, 1989.

JOURNEL, A. G.; HUIJBREGTS, C. J. *Mining Geostatistics*. London: Academic Press, 1978.

LARSEN, R. Decomposition Using Maximum Autocorrelation Factor. *Journal of Chemometrics*, v. 16, n. 8:10, p. 427-435, 2002.

LEUANGTHONG, O.; DEUTSCH, C. V. Stepwise Conditional Transformation for Simulation of Multiple Variables. *Mathematical Geology*, v. 35, n. 2, p. 155-173, 2003.

LITTLE, R.J.A.; RUBIN, D.B. *Statistical Analysis with Missing Data*. 2nd ed. New York: Wiley, 2002.

MACHUCA-MORY, D.; DEUTSCH, C. Non Stationary Variograms Based on Continuously Varying Weights. *CCG Annual Report*, v. 9, n. 117, p. 1-12, 2007.

MACHUCA-MORY, D.; DEUTSCH, C. Non-stationary Geostatistical Modeling Based on Distance Weighted Statistics and Distributions. *Mathematical Geosciences*, v. 45, p. 31-48, 2013.

MACQUEEN, J. Some Methods for Classification and Analysis of Multivariate Observations. In: LE CAM, L. M.; NEYMAN, J. (eds.). *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probabilities*, v. 1. Berkeley: University of California Press, 1967. p. 281-297.

MATHERON, G. Principles of Geostatistics. *Economic geology*, v. 58, n. 8, p. 1246-1266, 1963.

MATHERON, G. *The Theory of Regionalised Variables and its Applications*. Les Cahiers du Centre de Morphologie Mathématique de Fontainebleau No. 5. Paris: École Nationale Supérieure des Mines, 1971.

OWUSU, S.; DAGDELEN, K. Critical Review of Mineral Resource Classification Techniques in the Gold Mining Industry. *Insights in Mining Science and Technology*, v. 1, n. 3, p. 74-79, 2019.

PEARSON, K. On Lines and Planes of Closest Fit to Systems of Points in Space. *Philosophical Magazine*, v. 6, n. 2:11, p. 559–572, 1901.

ROBBINS, A.; BEEBE, N. H. F. *Classic Shell Scripting*. Sebastopol: O'Reilly Media, 2005.

SALDANHA, A. A. *Definição de Estratégia de Sondagem para Aumento na Conversão de Recursos e Reservas*. Dissertação (Mestrado em Engenharia), Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2020.

SILVA, C. J. E. *Influência do Espaçamento da Malha de Sondagem na Incerteza do Recurso Mineral*. Dissertação (Mestrado em Engenharia), Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2020.

SILVA, D. S. F. *Mineral Resource Classification and Drill Hole Optimization Using Novel Geostatistical Algorithms with a Comparison to Traditional Techniques*. Thesis (Master of Science in Mining Engineering), Department of Civil and Environmental Engineering. University of Alberta, Edmonton, 2015.

SINCLAIR, A. J.; BLACKWELL, G. H. *Applied Mineral Inventory Estimation*. Cambridge: University Press, 2004.

SOUZA, L. E.; COSTA, J. F. C. L.; KOPPE, J. C. Comparative Analysis of the Resource Classification Techniques: Case Study of the Conceição Mine, Brazil. *Applied Earth Science*, v. 119, n. 3, p. 165-175, 2010.

SWITZER, P.; GREEN, A. *Min/max Autocorrelation Factors for Multivariate Spatial Imagery*. Stanford: Department of Statistics, Stanford University, 1984. Technical Report N° 6.

VANN, J.; SANS, H. Global Estimation and Change of Support at the Enterprise Gold Mine, Pine Creek, Northern Territory – Application of the Geostatistical Discrete Gaussian Model. *In: APCOM XXV 1995 Conference*. Brisbane, 1995. p. 171-179.

VERLY, G.W. Sequential Gaussian Cosimulation: A Simulation Method Integrating Several Types of Information. *In: SOARES, A. (ed.). Geostatistics Tróia '92. Quantitative Geology and Geostatistics, v. 1*. Dordrecht: Kluwer Academic Publishers, 1993. p. 543-554.

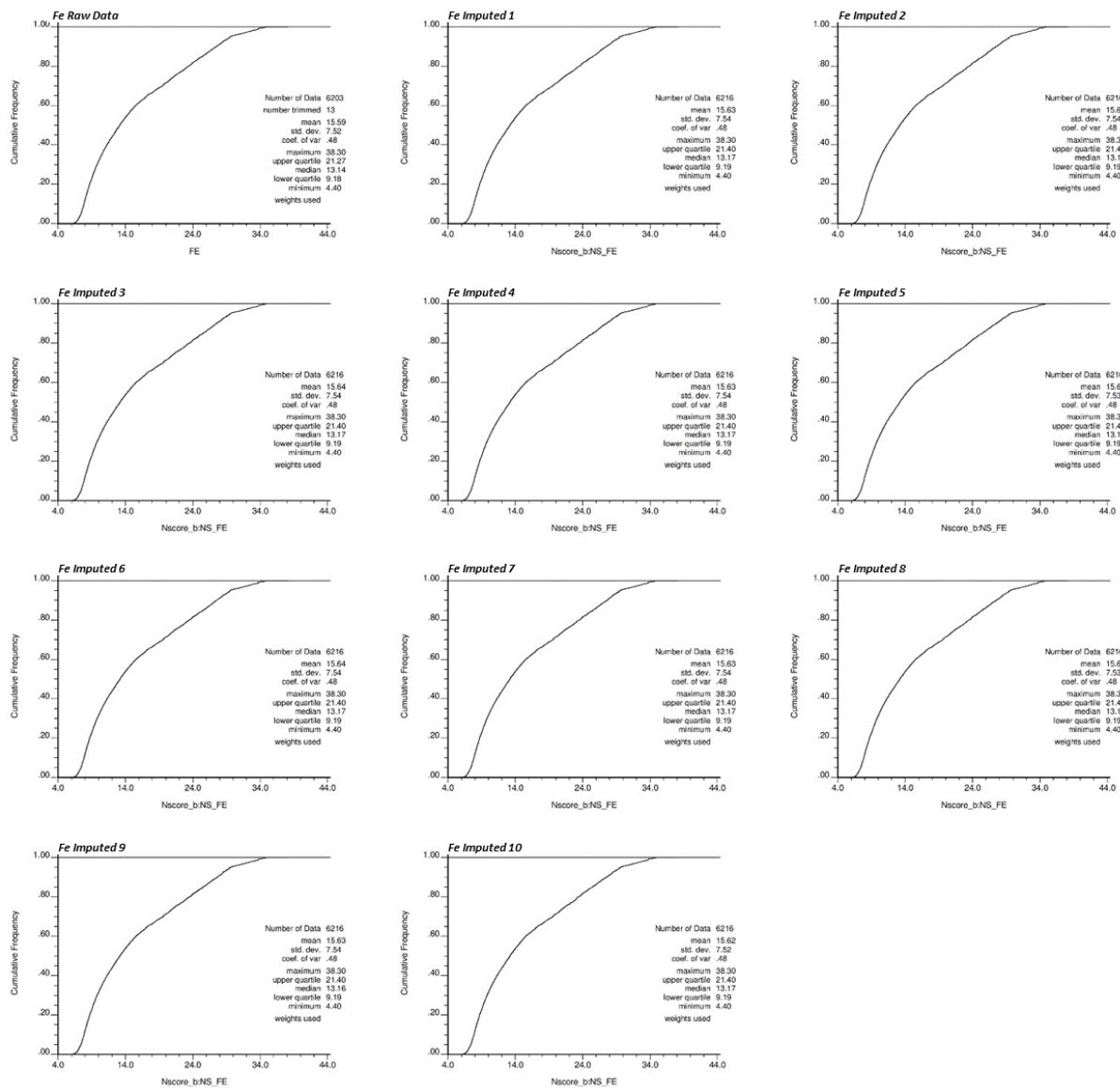
WACKERNAGEL, H. *Multivariate Geostatistics: An Introduction with Applications*. 2nd ed. Berlin: Springer, 1998.

WILDE, B. J.; DEUTSCH, C. V. A Comparison of Z Variograms Obtained by Transformation Using Hermite Polynomials and Monte Carlo Simulation. *CCG Annual Report*, v. 9, n. 407, p. 1-8, 2007.

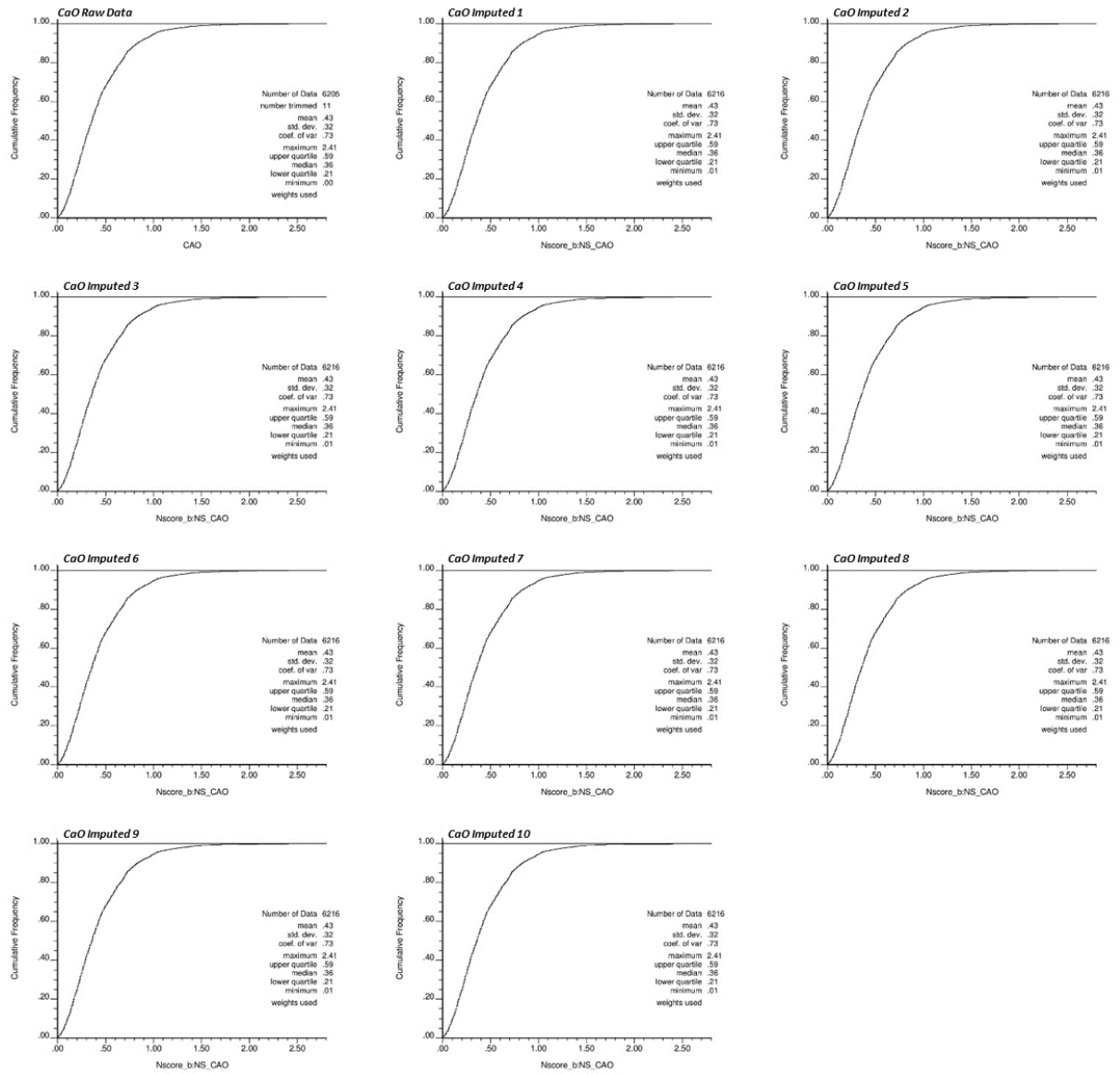
WILDE, B. J.; DEUTSCH, C. V. A New Approach to Calculate a Robust Variogram for Volume Variance Calculations and Kriging. *CCG Annual Report*, v. 7, n. 308, p. 1-11, 2005.

## APÊNDICE A – PREENCHIMENTO DE DADOS FALTANTES PARA FERRO E CÁLCIO

Assim como o MgO foi preenchido nos locais faltantes por meio da técnica de *data imputation*, conforme visto na seção 3.2, também foi feito preenchimento de dados para Fe e CaO, a fim de deixar a base de dados isotópica. Para estas variáveis também foram geradas dez bases de dados isotópicas e uma das bases de dados foi selecionada aleatoriamente para se prosseguir com a simulação. As figuras 26 e 27 contêm os resultados do preenchimento de dados faltantes para estas duas variáveis.



**Figura 26: Histogramas acumulados para Fe com dados isotópicos preenchidos (Imputed) e comparação com dados originais (Raw data).**



**Figura 27: Histogramas acumulados para CaO com dados isotópicos preenchidos (Imputed) e comparação com dados originais (Raw data).**

## APÊNDICE B – CALIBRAÇÃO DOS PARÂMETROS

Neste apêndice são apresentados alguns resultados de testes intermediários, os quais foram realizados a fim de calibrar melhor os parâmetros para o estudo de caso.

Os parâmetros que foram modificados em cada teste foram:

- Modelo de blocos (nós de simulação) na posição real ou projetada a um plano (*flattening*), considerando que a área de estudo possui uma leve inclinação de oeste para leste;
- Número de amostras usadas na simulação;
- Raio de busca;
- Uso de modelos variográficos com base nos fatores PPMT ou nscores;
- Modelo considerando apenas uma área menor (pit quinquenal) ou área total de estudo.

A Tabela 3 contém um resumo dos parâmetros que foram aplicados em cada teste. Os resultados para validação das reproduções dos histogramas e variogramas estão plotados da figura 28 a 39 (testes 1 a 6). O teste 7 foi o considerado aceito e seus resultados estão exibidos no capítulo 3 Estudo de Caso.

**Tabela 3: Resumo dos parâmetros aplicados em cada teste.**

Teste	Posição	Nº amostras	Busca XYZ (m)	Variogramas	Modelo
1	Real	40	150x150x30	Nscores	pit 5 anos
2	Real	60	200x200x30	Nscores	pit 5 anos
3	Real	24	100x100x30	Nscores	pit 5 anos
4	Projetada	40	150x150x30	Nscores	pit 5 anos
5	Real	40	150x150x30	PPMT factors	Total
6	Real	40	150x150x30	Nscores	Total
7	Real	40	150x150x30	Nscores - Continuidade calibrada	Total

Para efeitos comparativos, nos testes 2, 3, 4 e 6 foram plotados também os variogramas dos dados modelados diretamente sobre os valores originais (espaço real), além dos variogramas dos dados obtidos indiretamente, a partir da retro-transformação dos variogramas modelados sobre os dados transformados para o espaço gaussiano (nscores). Nota-se que os primeiros se mostram mais instáveis, por vezes com aspecto de efeito pepita puro, enquanto pelo segundo método os variogramas tendem a ter uma melhor definição.

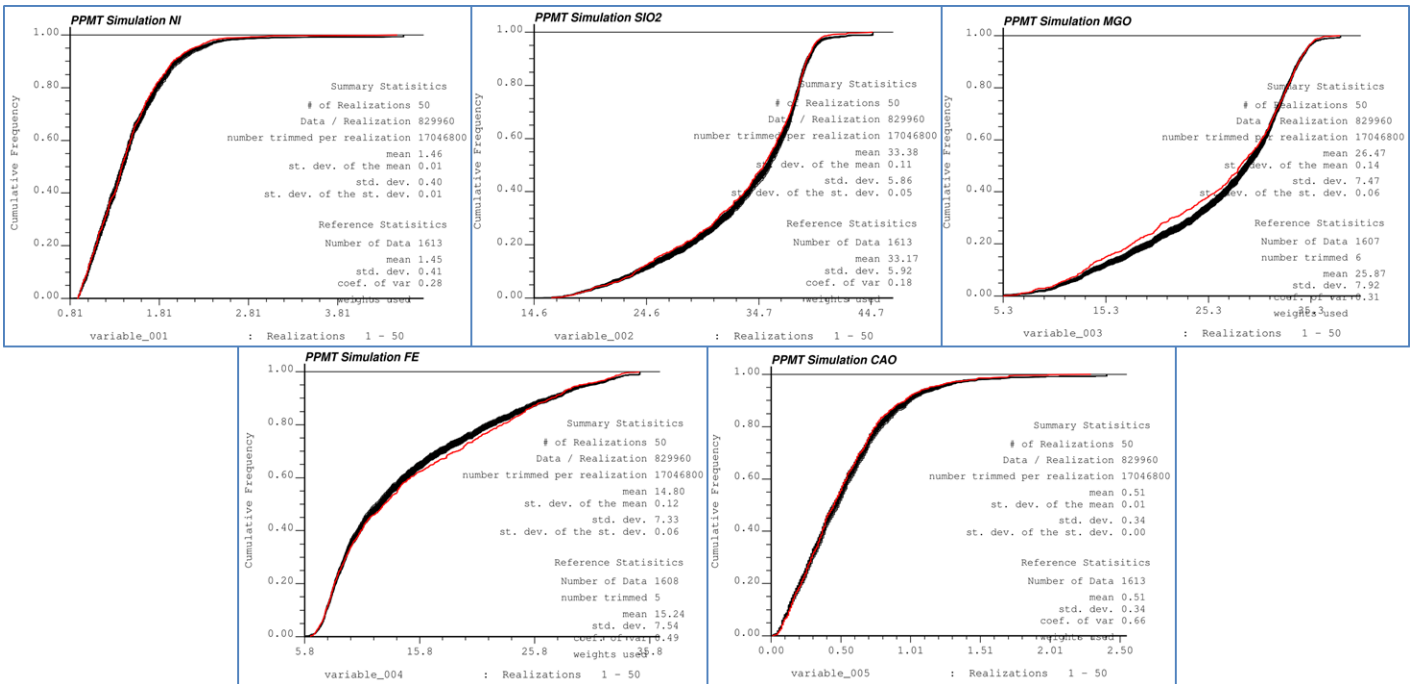


Figura 28: Teste 1 - Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto).

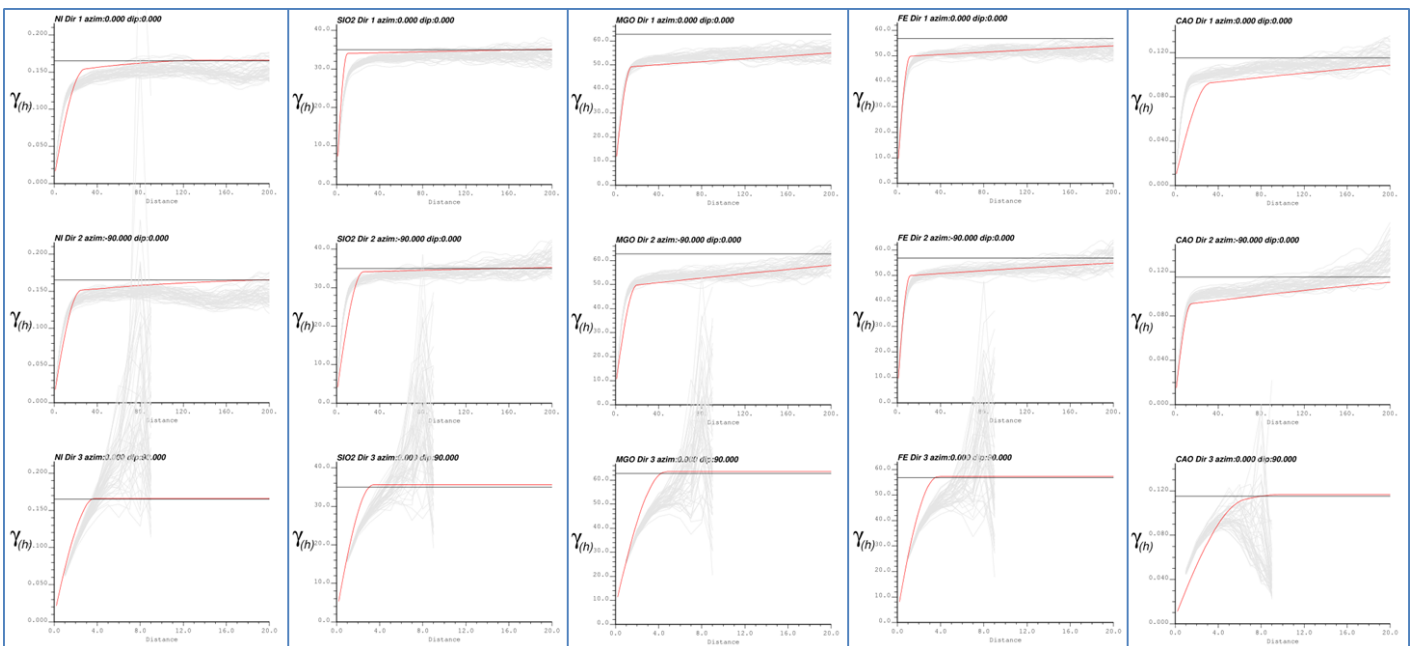


Figura 29: Teste 1 - Variogramas direcionais dos dados (vermelho) e das realizações simuladas (cinza). Cada linha representa uma direção e cada coluna representa uma variável de estudo.

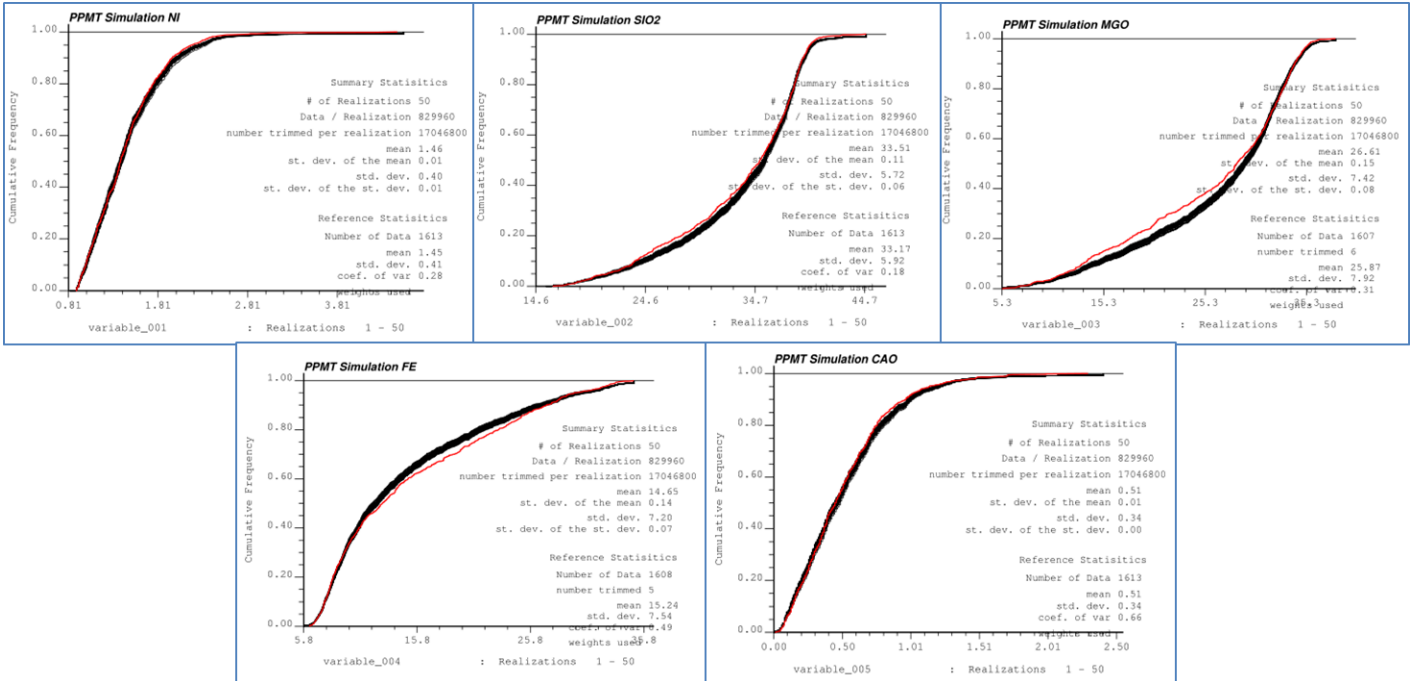


Figura 30: Teste 2 - Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto).

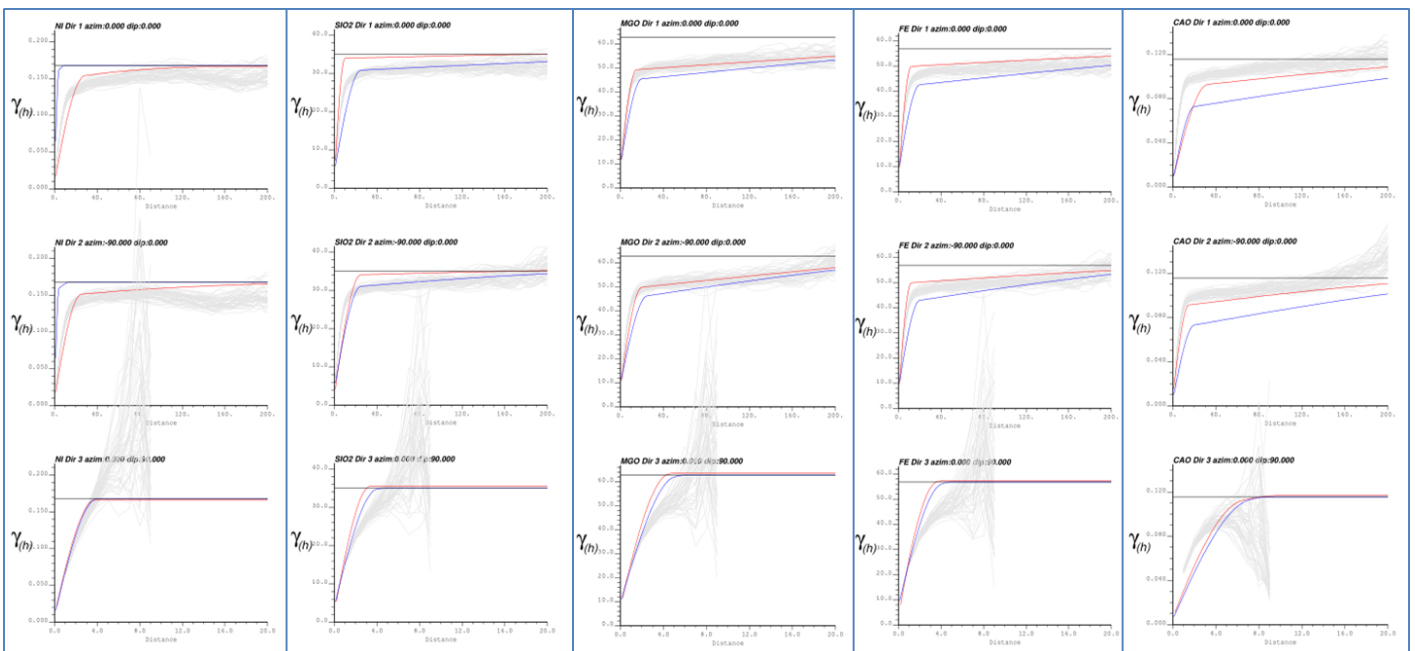


Figura 31: Teste 2 - Variogramas direcionais dos dados (vermelho: nscore retro-transformado; azul: dados originais) e das realizações simuladas (cinza). Cada linha representa uma direção e cada coluna representa uma variável de estudo.



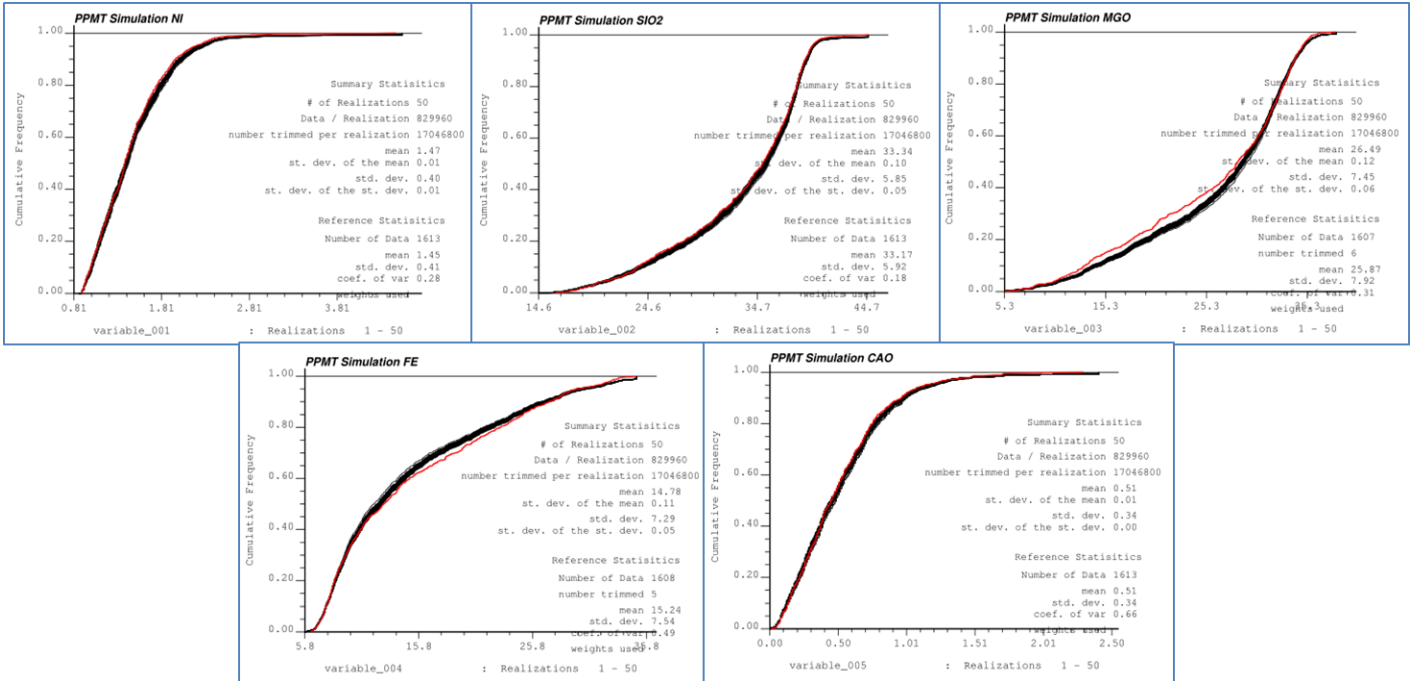


Figura 32: Teste 3 - Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto).

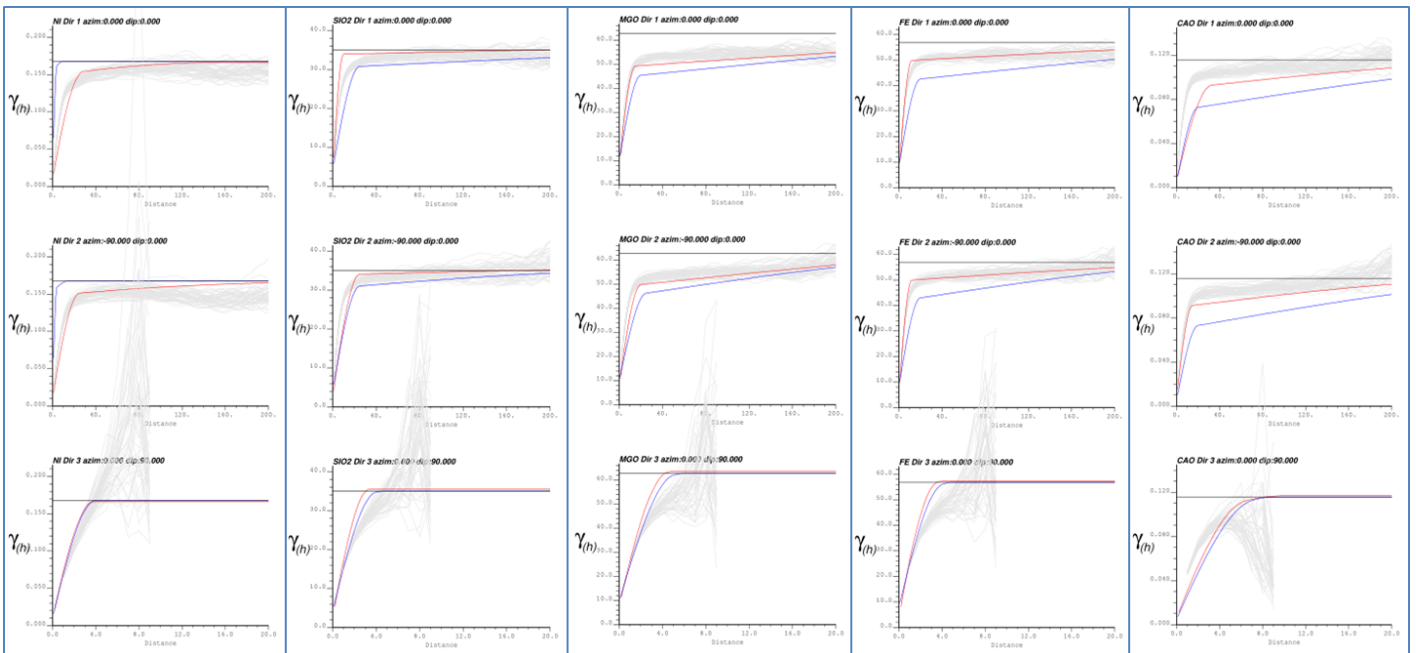


Figura 33: Teste 3 - Variogramas direcionais dos dados (vermelho: nscore retro-transformado; azul: dados originais) e das realizações simuladas (cinza). Cada linha representa uma direção e cada coluna representa uma variável de estudo.

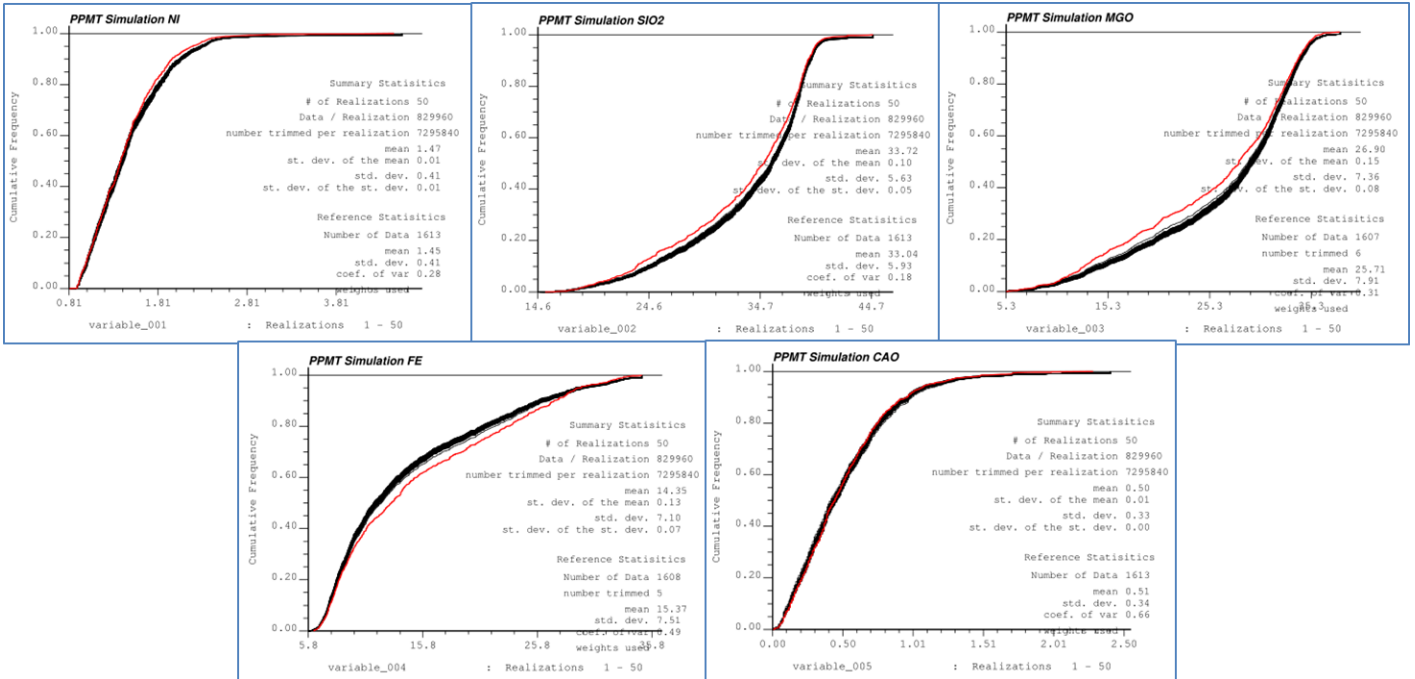


Figura 34: Teste 4 - Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto).

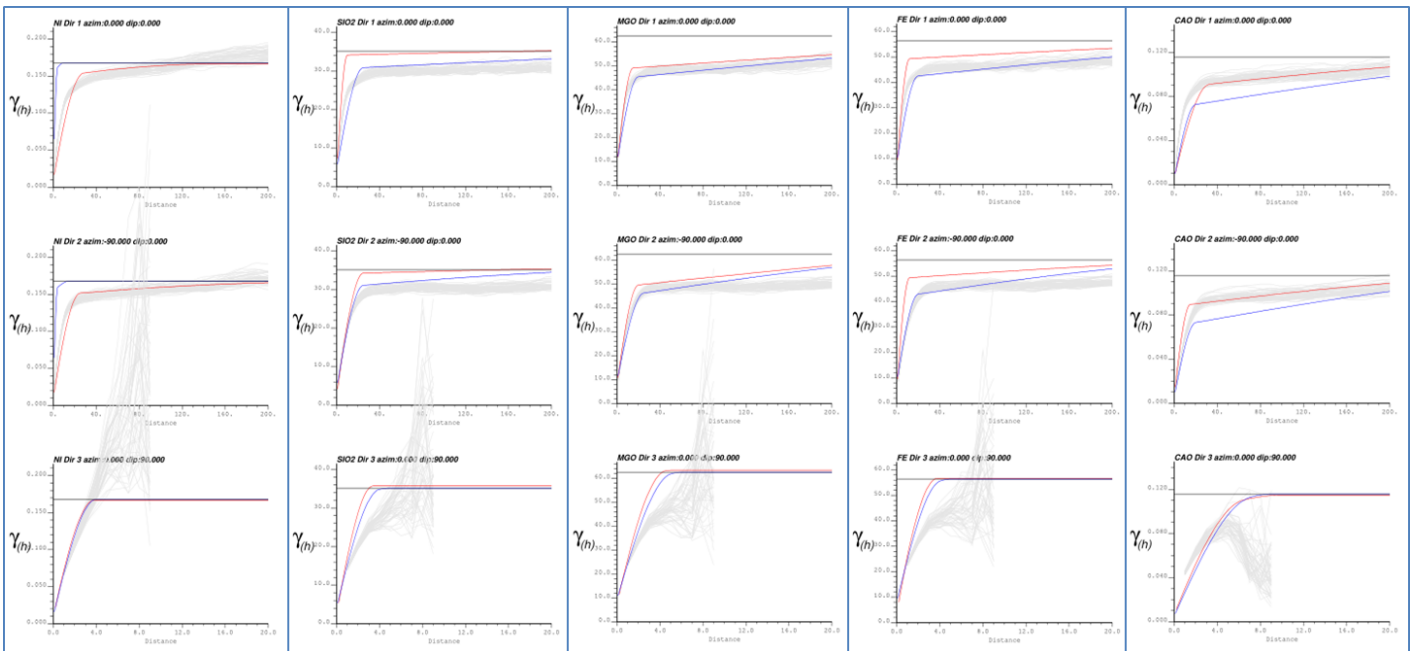
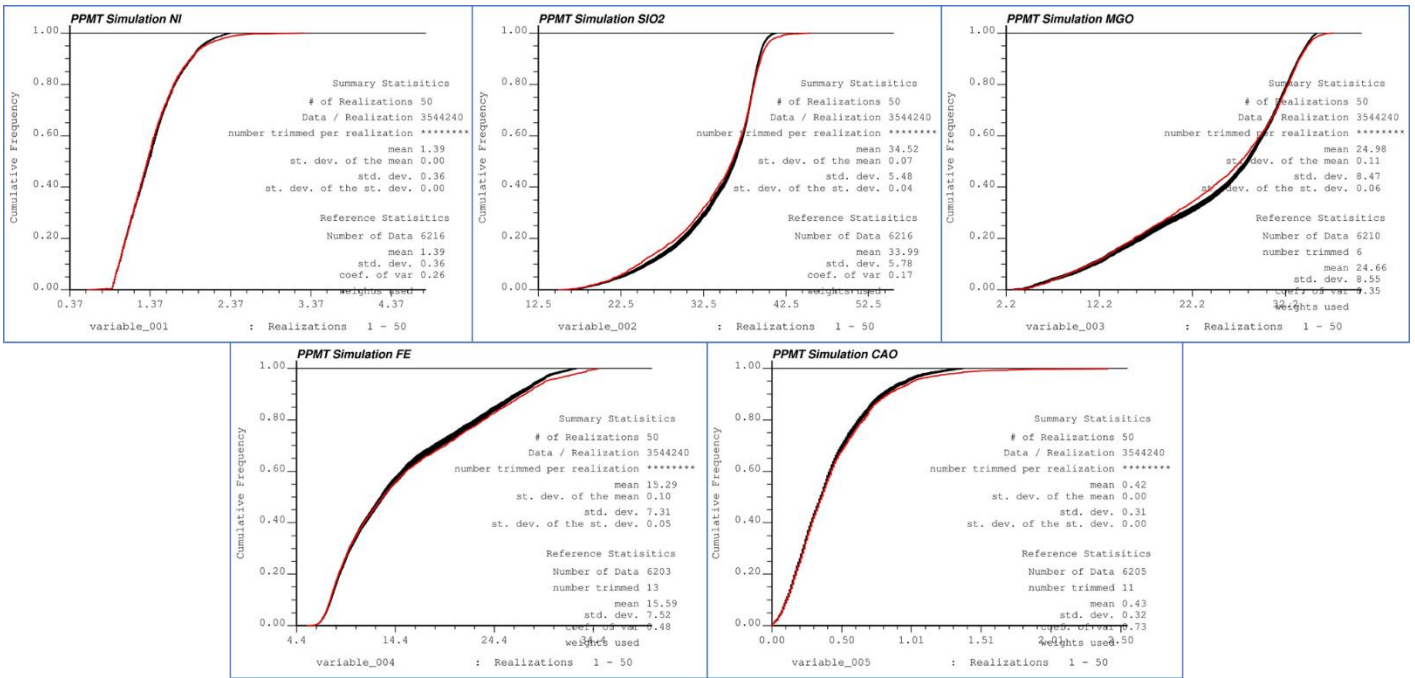
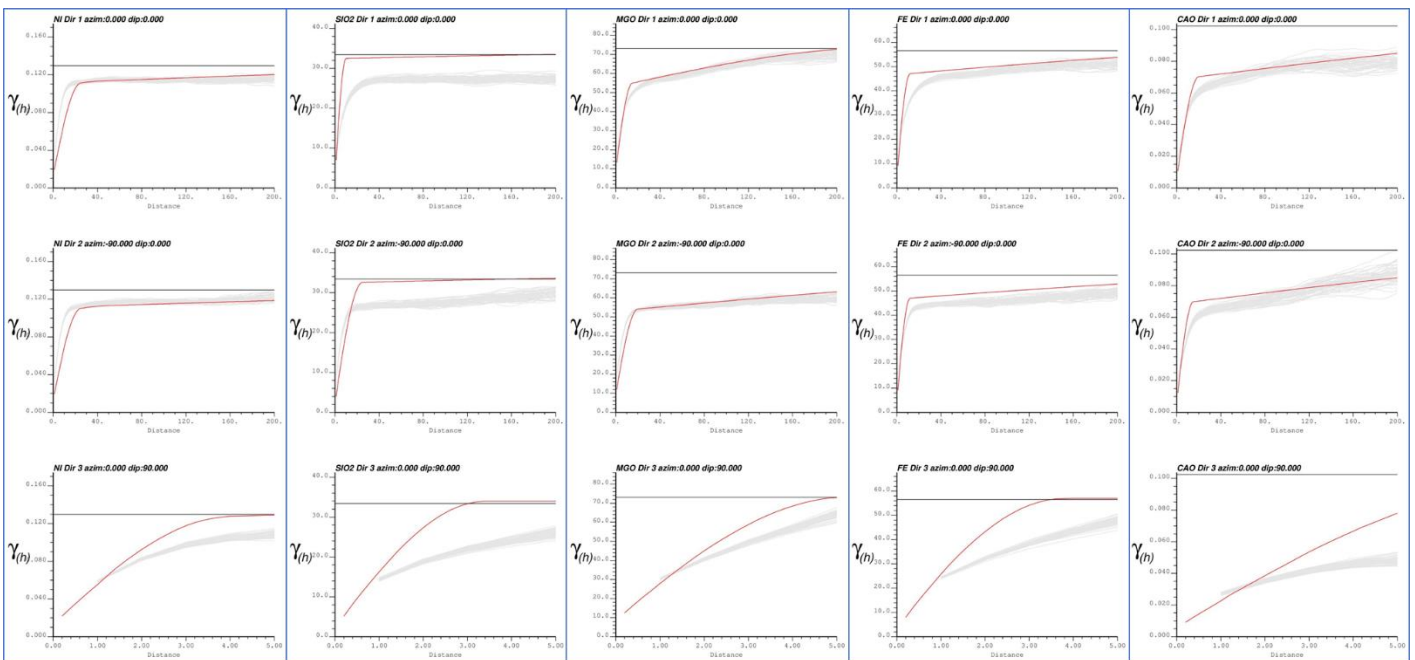


Figura 35: Teste 4 - Variogramas direcionais dos dados (vermelho: nscore retro-transformado; azul: dados originais) e das realizações simuladas (cinza). Cada linha representa uma direção e cada coluna representa uma variável de estudo.



**Figura 36: Teste 5 - Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto).**



**Figura 37: Teste 5 - Variogramas direcionais dos dados (vermelho: nscore retro-transformado; azul: dados originais) e das realizações simuladas (cinza). Cada linha representa uma direção e cada coluna representa uma variável de estudo.**

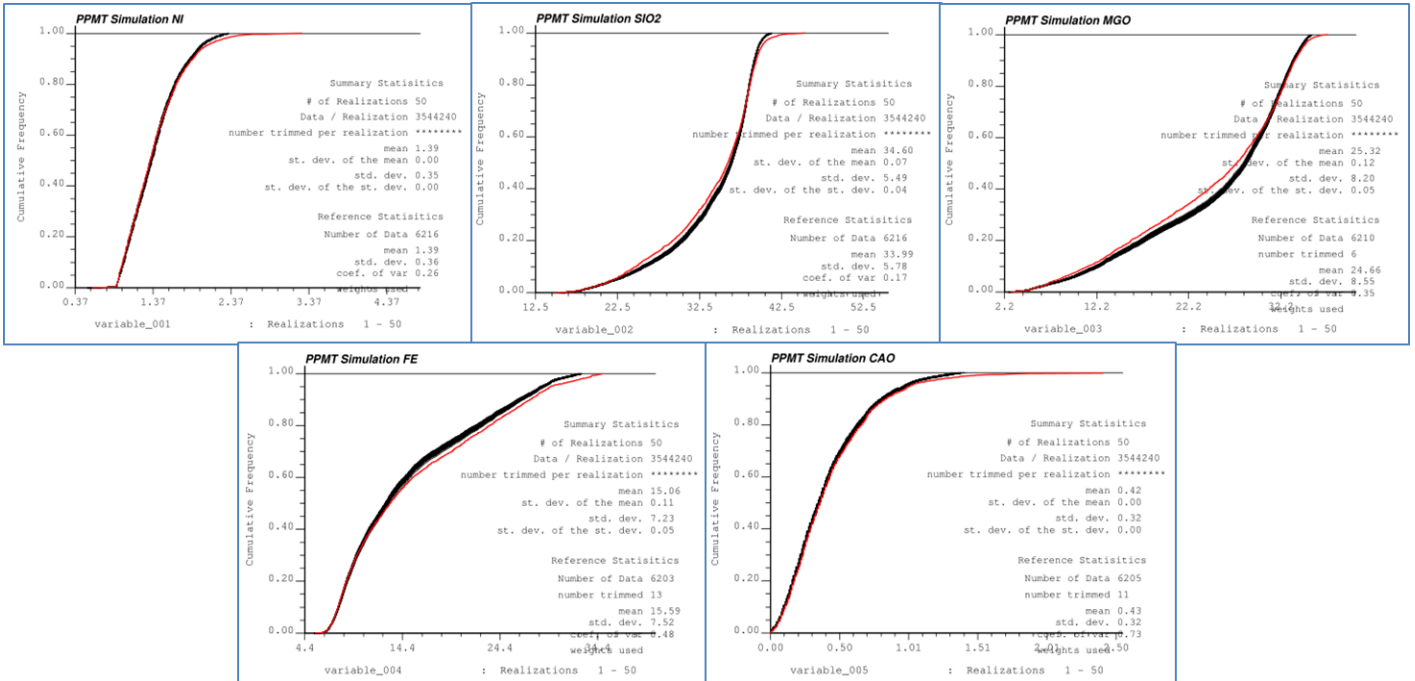


Figura 38: Teste 6 - Histogramas dos dados desagrupados (vermelho) e das realizações simuladas (preto).

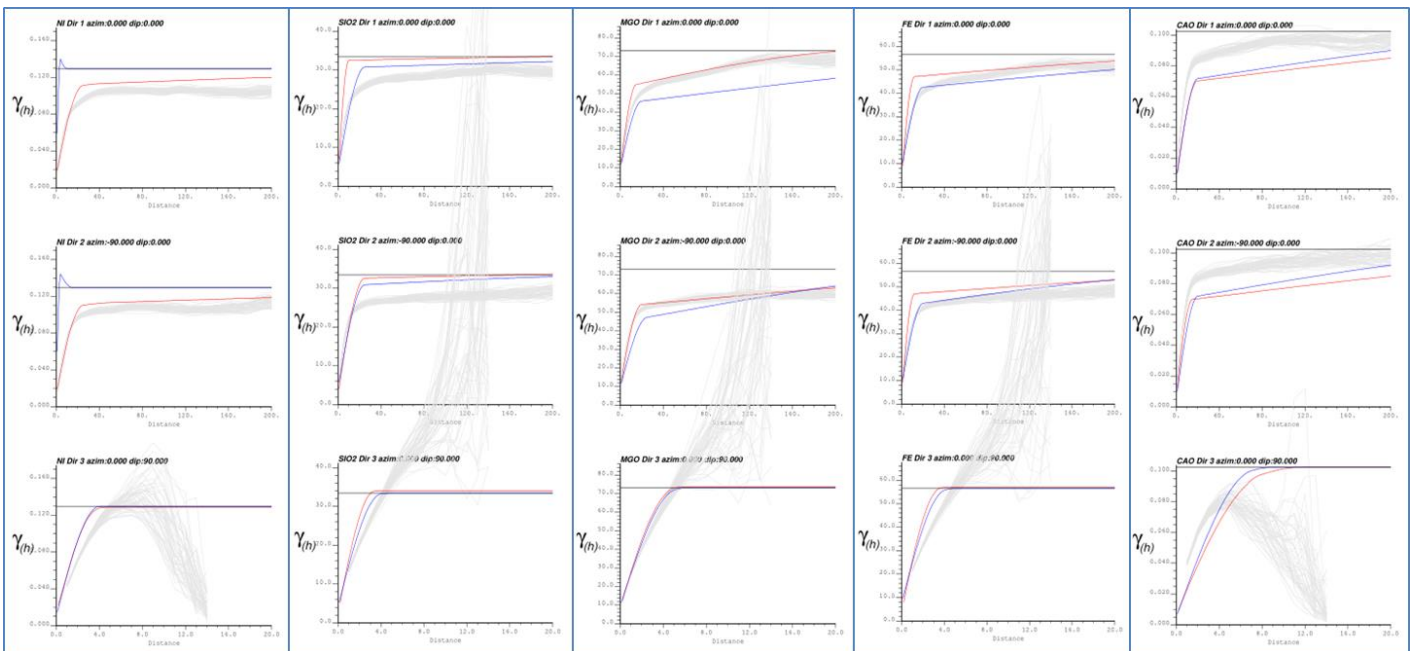


Figura 39: Teste 6 - Variogramas direcionais dos dados (vermelho: nscore retro-transformado; azul: dados originais) e das realizações simuladas (cinza). Cada linha representa uma direção e cada coluna representa uma variável de estudo.

Como última etapa de calibração, os variogramas do Teste 6 tiveram suas continuidades espaciais reduzidas ou infladas para tentar minimizar os efeitos de ganho ou perda da continuidade espacial durante a simulação, a fim de refletir melhor os variogramas dos dados (Figura 40).

*NS Ni*

$$\begin{aligned}\gamma_0(h) &= 0.10 + 0.75\text{Sph}_h(25\text{m}) + 0.15\text{Sph}_h(600\text{m}) \\ \gamma_{90}(h) &= 0.10 + 0.75\text{Sph}_h(25\text{m}) + 0.15\text{Sph}_h(750\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.10 + 0.75\text{Sph}_h(4.5\text{m}) + 0.15\text{Sph}_h(5\text{m})\end{aligned}$$

*NS SiO<sub>2</sub>*

$$\begin{aligned}\gamma_0(h) &= 0.05 + 0.90\text{Sph}_h(10\text{m}) + 0.05\text{Sph}_h(400\text{m}) \\ \gamma_{90}(h) &= 0.05 + 0.90\text{Sph}_h(25\text{m}) + 0.05\text{Sph}_h(350\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.05 + 0.90\text{Sph}_h(3.4\text{m}) + 0.05\text{Sph}_h(3.5\text{m})\end{aligned}$$

*NS MgO*

$$\begin{aligned}\gamma_0(h) &= 0.10 + 0.60\text{Sph}_h(15\text{m}) + 0.30\text{Sph}_h(250\text{m}) \\ \gamma_{90}(h) &= 0.10 + 0.60\text{Sph}_h(20\text{m}) + 0.30\text{Sph}_h(600\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.10 + 0.60\text{Sph}_h(5\text{m}) + 0.30\text{Sph}_h(6\text{m})\end{aligned}$$

*NS Fe*

$$\begin{aligned}\gamma_0(h) &= 0.05 + 0.75\text{Sph}_h(12\text{m}) + 0.20\text{Sph}_h(400\text{m}) \\ \gamma_{90}(h) &= 0.05 + 0.75\text{Sph}_h(12\text{m}) + 0.20\text{Sph}_h(470\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.05 + 0.75\text{Sph}_h(3.7\text{m}) + 0.20\text{Sph}_h(4.0\text{m})\end{aligned}$$

*NS CaO*

$$\begin{aligned}\gamma_0(h) &= 0.05 + 0.60\text{Sph}_h(20\text{m}) + 0.35\text{Sph}_h(600\text{m}) \\ \gamma_{90}(h) &= 0.05 + 0.60\text{Sph}_h(15\text{m}) + 0.35\text{Sph}_h(600\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.05 + 0.60\text{Sph}_h(8\text{m}) + 0.35\text{Sph}_h(12\text{m})\end{aligned}$$

*NS Ni Modificado*

$$\begin{aligned}\gamma_0(h) &= 0.05 + 0.90\text{Sph}_h(13\text{m}) + 0.05\text{Sph}_h(400\text{m}) \\ \gamma_{90}(h) &= 0.05 + 0.90\text{Sph}_h(16\text{m}) + 0.05\text{Sph}_h(500\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.05 + 0.90\text{Sph}_h(2\text{m}) + 0.05\text{Sph}_h(3\text{m})\end{aligned}$$

*NS SiO<sub>2</sub> Modificado*

$$\begin{aligned}\gamma_0(h) &= 0.03 + 0.95\text{Sph}_h(6\text{m}) + 0.02\text{Sph}_h(100\text{m}) \\ \gamma_{90}(h) &= 0.03 + 0.95\text{Sph}_h(10\text{m}) + 0.02\text{Sph}_h(80\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.03 + 0.95\text{Sph}_h(1\text{m}) + 0.02\text{Sph}_h(2\text{m})\end{aligned}$$

*NS MgO Modificado*

$$\begin{aligned}\gamma_0(h) &= 0.15 + 0.45\text{Sph}_h(8\text{m}) + 0.40\text{Sph}_h(250\text{m}) \\ \gamma_{90}(h) &= 0.50 + 0.45\text{Sph}_h(15\text{m}) + 0.40\text{Sph}_h(500\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.15 + 0.45\text{Sph}_h(5\text{m}) + 0.40\text{Sph}_h(6\text{m})\end{aligned}$$

*NS Fe Modificado*

$$\begin{aligned}\gamma_0(h) &= 0.10 + 0.85\text{Sph}_h(12\text{m}) + 0.05\text{Sph}_h(500\text{m}) \\ \gamma_{90}(h) &= 0.10 + 0.85\text{Sph}_h(13\text{m}) + 0.05\text{Sph}_h(400\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.10 + 0.85\text{Sph}_h(3\text{m}) + 0.05\text{Sph}_h(4.5\text{m})\end{aligned}$$

*NS CaO Modificado*

$$\begin{aligned}\gamma_0(h) &= 0.15 + 0.55\text{Sph}_h(40\text{m}) + 0.30\text{Sph}_h(300\text{m}) \\ \gamma_{90}(h) &= 0.15 + 0.55\text{Sph}_h(35\text{m}) + 0.30\text{Sph}_h(250\text{m}) \\ \gamma_{\text{vert}}(h) &= 0.15 + 0.55\text{Sph}_h(6\text{m}) + 0.30\text{Sph}_h(15\text{m})\end{aligned}$$

**Figura 40: Calibração dos variogramas dos nscores referentes à cada variável. Teste 6 à esquerda e Teste 7 à direita.**

## APÊNDICE C – E-TYPE VS. KRIGAGEM ORDINÁRIA

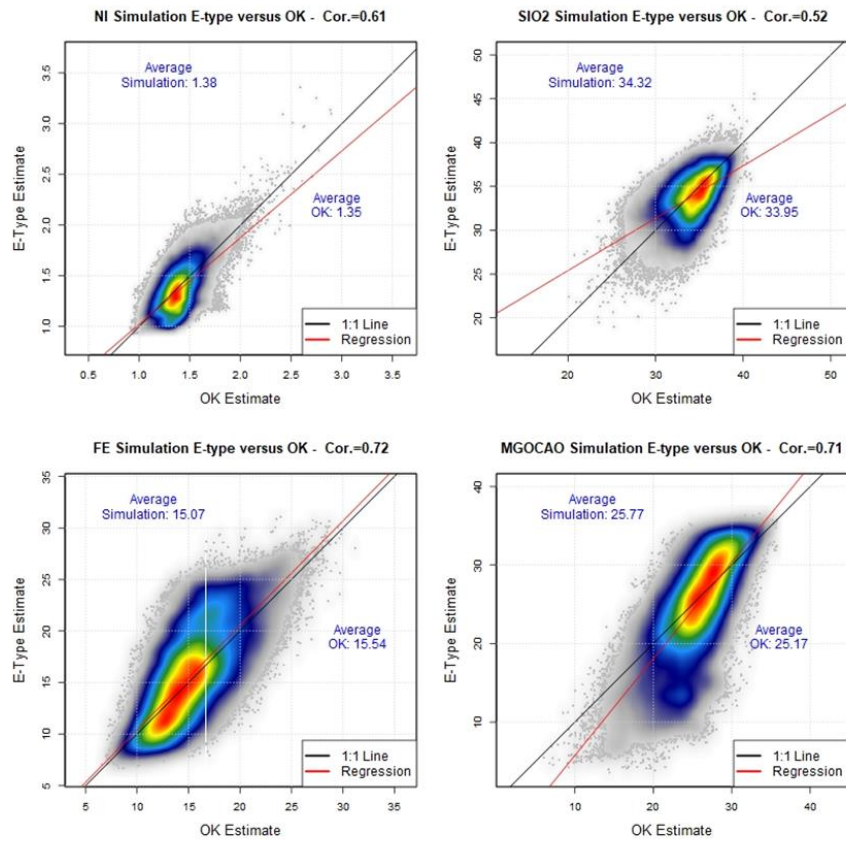
Para efeitos comparativos e discussão com o planejamento de mina, a média das realizações (E-type) foi comparada com as estimativas do modelo em vigor no depósito, as quais são feitas por krigagem ordinária.

Devido à uma limitação do número de colunas permitidas no software utilizado na mina em questão, a variável CaO foi agrupada à variável MgO criando uma única variável MgOCaO.

Os resultados comparativos mostram boas correlações para Fe e MgOCaO e correlações intermediárias para Ni e SiO<sub>2</sub>. Apesar disso, existe uma correlação positiva entre a média das realizações (E-type) e a krigagem ordinária para todas as variáveis, como era de se esperar, e as diferenças entre os valores médios não são considerados excessivos (Figura 41).

A relativa elevada dispersão dos valores pode sugerir que as estimativas por krigagem ordinária não estão bem condicionadas, com um grau de liberdade relativamente alto, resultando em uma ampla gama de teores estimados. Também, apesar das diferenças entre a média das realizações E-type e estimativas por krigagem ordinária serem pequenas, o fato de que a KO não está compreendida dentro do intervalo (mínimo e máximo) das realizações pode ser devido à pouca variação obtida nas realizações (flutuações ergódicas), como se pode notar na checagem da reprodução dos histogramas. Para melhorar a convergência deste resultado, algumas melhorias podem ser implementadas em ambos os métodos, como uso de diferentes sementes e bases de dados isotópicas para diferentes realizações simuladas, assim como alterações de parâmetros de busca na krigagem, para melhorar o condicionamento das amostras e reduzir o viés.

Além destes pontos, outros motivos para as diferenças são: 1. A simulação usou krigagem simples; 2. A krigagem ordinária usou uma limitação da influência espacial dos valores extremos.



	Kt	Ni	Fe	SiO <sub>2</sub>	CaOMgO	
OK Estimation	6,211.87	1.37	15.38	34.14	25.32	
Minimum Simulation	6,189.77	1.38	14.94	34.16	25.48	
Maximum Simulation	6,197.83	1.40	15.33	34.42	25.92	
Average Simulation E-Type	6,193.95	1.39	15.10	34.31	25.74	
Diff. OK/E-type		0.3%	-1.7%	1.9%	-0.5%	-1.6%

**Figura 41: Comparação entre o E-type versus a krigagem ordinária.**

## APÊNDICE D – BASH SCRIPTS

Os scripts *bash* foram divididas em dez etapas:

- Sim0\_Rockmodel.bsh: criação do protótipo mínimo para simulação em formato gsb;
- Sim1\_EDA.bsh: análise exploratória dos dados;
- Sim2.3\_Dataprep.bsh: preparação dos dados com transformação nscores e variogramas;
- Sim3.2\_Impute.bsh: preenchimento de dados faltantes e transformação PPMT;
- Sim4.1\_Simulate.bsh: simulação sequencial gaussiana;
- Sim5\_Backtransform.bsh: retro-transformação das realizações simuladas para o espaço real;
- Sim6\_Sim\_Histograms.bsh: histogramas dos dados e realizações para validação;
- Sim7A\_Sims\_Variograms.bsh: cálculo dos variogramas das realizações;
- Sim7C\_Plot\_Variograms\_vertical5m.bsh: plotagem dos variogramas das realizações e dos dados para validação;
- Sim8.1\_Ascii\_output.bsh: exportação dos resultados para formato “.csv”.

Os códigos para execução dos scripts em ambiente *bash* estão disponíveis neste apêndice. A página seguinte contém um arquivo de parâmetros geral, contendo os parâmetros principais para execução dos scripts.



```

=====
# Bash script system: PPMT simulation
# Parameter File Listing to control script runs
=====

#input data file and column numbers for nvar variables. var(nvar) should be last column of file.
input_data="dh5ppmt_total.csv"
xcol=1
ycol=2
zcol=3
nvar=5
var1=7
var2=8
var3=9
var4=10
var5=11

#number of realisations to create
nrealisations=50

#number of data values to include in simulation
ndata_sim=40

#angles for simulation Search ellipsoid [use "" to enclose 3 variables as one]
search=(0.0, 0.0, 0.0)
radii=(150, 150, 30)

#Geological model is name of Gslib-format file with nodes to simulate. Keep_code = litho code
input_nodes=md5ppmt_total.csv
keep_code=200

#Node spacing in geological model data
xsiz=1.25
ysiz=1.25
zsiz=1.0

#Output rock models [1 realisation in first model, nrealisations in second model]. Use *.gsb
rock=rockmod.gsb
rock_nreal=rockmod_nreal.gsb

#Output grid definition [ascii]
grid_definition_file=simgrid

#Orientations of variogram vectors in the simulated grid [calculate these to closely match
variograms directions]
ixd_1=0
iyd_1=1
izd_1=0
ixd_2=1
iyd_2=0
izd_2=0
ixd_3=0
iyd_3=0
izd_3=-1

#Random number seed
random_seed=5514131

#END of Parameter List

```

```

#!/bin/bash
#=====
# Script Sim0_Rockmodel.bsh
#=====
# Input a Gslib file of coordinates and a Rockcode to define the Simulation Target Volume
#=====

#read Master_Parameter List

echo "Sim0_Rockmodel.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    chmod 777 PPMT_Master_Parameters.txt
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
#source the parameter file
. PPMT_Master_Parameters.txt

#create some subdirectories to store results
echo making subdirectories

mkdir -p eda; chmod 777 eda
mkdir -p vgram; chmod 777 vgram
mkdir -p ppmt; chmod 777 ppmt
mkdir -p sims; chmod 777 sims
mkdir -p nscore; chmod 777 nscore
mkdir -p rock; chmod 777 rock
mkdir -p work; chmod 777 work

#name of rockmodel data file
infile=$input_nodes

#rock code in model
keepcode=$keep_code

#how many realisations will we make?
nreal=$nrealisations

#names of output files:
grid=$grid_definition_file
outmod=$rock
gsbmodnreal=$rock_nreal

#node spacing in model
xinc=$xsiz
yinc=$ysiz
zinc=$zsiz

```

```

#-----
# Find the Smallest Grid we can use - only enter the cell sizes
#-----
maxcode=$((keepcode+1))
mincode=$((keepcode-1))
date
cat<<END>>temp

                Parameters for ROCKMOD
                *****

START OF PARAMETERS:
$infile          -file with data
1 2 3 4          -columns for X,Y,Z, var.
-999.0 1.0E21    -trimmig limits on var 1
rock/$grid       -output file
rockmod.dbg      -debug file
1 0.0 $xinc      -X grid specification
1 0.0 $yinc      -Y grid specification
1 0.0 $zinc      -Z grid specification
0                -0=output grid 1=output rockmod
END

echo temp | Rockmod
rm temp rockmod.dbg

#-----
# Now rerun this using the Grid defined to get the Rock Model
#-----

cat<<END>>temp

                Parameters for ROCKMOD
                *****

START OF PARAMETERS:
$infile          -file with data
1 2 3 4          -columns for X,Y,Z, var.
$mincode $maxcode -trimming limits on var 1
rock/$outmod     -output file
rock/rockmod.dbg -debug file
END

cat rock/$grid>>temp

cat<<END>>>temp
1                -0=output grid 1=output rockmod
END

echo temp | Rockmod
rm temp

date
#-----
# Write rockmodel in gsb format
#-----
# get the number of grid nodes
awk 'NR==1{print $1}' rock/$grid > work/nx
awk 'NR==2{print $1}' rock/$grid > work/ny
awk 'NR==3{print $1}' rock/$grid > work/nz

```

```

#-----
# Now extend this grid nreal times : we need the Ascii and Binary Versions for now
#-----

cat<<END>temp

Parameters for UMERGE_MANIP
*****

START OF PARAMETERS:
rock/$gsbmodnreal          -output file
$(work/nx) $(work/ny) $(work/nz) $nreal -nx, ny, nz, nreal (calculated if nx*ny*nz=0)
1 $maxcode                 -trimming limits
1                          -# of gridded models to merge
rock/$outmod               -file(1)
1 1                        -# of variables and col #s to extract
$nreal                    -replicate single grid nreal times?
END

echo temp | umerge_manip
rm temp
rm work/nx; rm work/ny; rm work/nz

date

#-----
# Check: count the number of occurrences of 244 in outmod
#-----

echo Checking number of blocks...
awk /$keepcode/ $infile | wc | awk '{print $1}' > count
echo There are $(count) data lines in $infile
rm count

cat<<END>temp

                Parameters for GSB_PREVIEW
                *****

START OF PARAMETERS:
rock/$outmod          -file with binary data
0 0                  -first and last realization to display? (0=all)
5                    -number of records to display per realization (0=10)
0                    -output file (1) or command window (0)?
gsb_preview.txt      -if 1, name of the output file
END

echo temp | gsb_preview; rm temp

echo "Sim0_Rockmodel.bsh complete"

#=====
# Script complete
#=====

#*****

```

```

#!/bin/bash
#=====
# Script Sim1_EDA.bsh
#=====
# Exploratory Data Analysis
#=====

#read Master_Parameter List

echo "Sim1_EDA.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    chmod 777 PPMT_Master_Parameters.txt
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
#source the parameter file
. PPMT_Master_Parameters.txt

#column numbers in input file
file=$input_data
nvar=$nvar
col1=$var1
col2=$var2
col3=$var3
col4=$var4
col5=$var5
wtcol=0

#what are the trimming limits on these variables?
trimlow=-0.0001
trimhigh=100

#these three outputs will be written
scatterout=eda/RawData_scatter.ps
histout=eda/RawData_histograms.ps
probout=eda/RawData_probability.ps

#-----
# Run casebuild for data summary
#-----

cat<<END>temp
                Parameters for CASEBUILD
                *****

START OF PARAMETERS:
$file                -file with data
$nvar $col1 $col2 $col3 $col4 $col5    -number of variables and columns
$trimlow $trimhigh  -trimming limits
work/settings.sh    -file for output case statement
END
echo temp |casebuild
chmod 777 work/settings.sh #make sure we have all access to the settings file
dos2unix work/settings.sh #make sure settings.sh has UNIX line endings

```

```

#-----
# Run scatterplots
#-----

runScatplt()
{
cat<<END >temp

                Parameters for SCATPLT
                *****

START OF PARAMETERS:
1                -number of data files to plot
$file            -main data file (will calculate statistics, default ...
${col1} ${col2} $wtcol 0 -columns for X, Y, wt, third var. ...
$trimlow $trimhigh -trimming limits ...
0 0 2 10 -1      -line width (0=none), dashing, bullet size (0=none), ...
${num}.ps        -file for Postscript output ...
2 1 0            -X min and max, (0=arith, 1=log) (max<min for auto ...
5 3 0            -Y min and max, (0=arith, 1=log) (max<min for auto ...
1                -plot every nth data point ...
0.0 20           -limits for third variable gray scale ...
1                -reference stats to plot(-1=none,0=cor only, 1=minimal...
0.8              -positioning of stats (L to R: -1 to 1) ...
3                -plot x/y histograms of the first data file (0=none, ...
${title}         -title ...
default          -X axis label (default=get from data file #1) ...
default          -y axis label (default=get from data file #1) ...
END

echo temp | scatplt2008
}

# loop controlling scatterplot
num=0
N=$nvar
for ((x=1 ; x<N ; x++)); do
let ys=x+1
for ((y=ys; y<=N; y++)); do
# X-Axis Variables
. work/settings.sh ${x}
col1=${col}
name1=${name}
# Y-Axis Variables
. work/settings.sh ${y}
col2=${col}
name2=${name}
# Common Variables
let num=num+1
title="${name1} vs ${name2}"

runScatplt

done
done

```

```

#-----
# Group all scatterplots
#-----

# Construct a plotem parameter file from standard input, as well as a
# for loop to populate the necessary postscripts and 'nofile' placeholders

cat<<END>temp
        Parameters for PLOTEM
        *****

START OF PARAMETERS:
$scatterout
END

echo "$((N-1)) $((N-1))" >>temp

num=0

for ((x=1 ; x<N ; x++)); do
    for ((y=2; y<=N; y++)); do

        if [ ${y} -gt ${x} ]; then
            let num=num+1
            echo "${num}.ps" >>temp
        else
            echo "nofile" >>temp
        fi

    done
done

plotem temp

rm temp ?.ps ??.ps

```

```

#-----
# Run histograms
#-----

runHistplt()
{
cat<<END >temp

                Parameters for HISTPLT
                *****

START OF PARAMETERS:
$file                -file with reference data
${col}  $wtcol        -columns for variable and weight (set ivr<0 to ...
$trimlow  $trimhigh  -trimming limits ...
${x}.ps          -file for PostScript output ...
0.0      -1          -attribute minimum and maximum (<0 for automatic ...
-1.0     -1.0        -frequency maximum (<0 for automatic) ...
35       0           -number of classes (<0 to label bars with number ...
0        0           -0=arithmetic, 1=log scaling ...
0        0           -0=frequency, 1=cumulative histogram, 2=constant...
20       -2          -number of cum. quantiles (<0 for all) or the ...
-2       -2          -number of decimal places (<0 for auto.) ...
${header}        -title ...
default          -X label, if set to default the variable in ...
Frequency        -Y label, if set to default the variable in ...
1e21            -reference value for box plot ...
1.5             -horizontal positioning of stats (L to R: -1 to ...
2              -reference stats to plot(0=none,1=minimal,2=ful ...
reals.out       -file for realization data (if plotting realizati...
1 0            -columns for variable and weight
10            -nreal
10 100 1      -nx ny nz
END

echo temp | histplt2008
}

N=$nvar
for ((x=1 ; x<=N ; x++)); do
    . work/settings.sh ${x}
    header="${name} PDF"
    runHistplt
done

```



```

#-----
# Group all histograms
#-----

cat<<END>temp

                Parameters for PLOTEM
                *****

START OF PARAMETERS:
$histout
2 3
END

for ((x=1 ; x<=N ; x++)); do

    echo "${x}.ps" >>temp
done

    plotem temp
    rm temp ?.ps ??.ps junk.out

#-----
# Run probability plots
#-----

runProbplt()
{
cat<<END >temp

                Parameters for PROBLPT
                *****

START OF PARAMETERS:
$file                -file with data                ...
${col}  $wtcol        -columns for variable and weight            ...
$trimlow  $trimhigh  -trimming limits                            ...
${x}.ps          -file for PostScript output            ...
0                -number of points to plot (<0 for all)        ...
0                -0=arithmetic, 1=log scaling                ...
0  ${max}  ${inc}    -min,max, increment for labeling (Xaxis)        ...
${header}        -title                                    ...
Cumulative Probability  -Yaxis label                            ...
default          -Xaxis label ( default to use label in data file)  ...
-1               -gridding (1=plot grid, 2=remove grid, neg number for...
0.75             -positioning of stats (L to R: -1 to 1)            ...
2 4              -level of detail in stats (from 0=none to 2=detailed)...
END

echo temp | probplt2008
}

N=$nvar
for ((x=1 ; x<=N ; x++)); do
    . work/settings.sh ${x}
    header="${name} Probability Plot"
    y=${max}
    inc=$(bc <<< "scale=2; $y / 5 " )
    runProbplt
done

```

```
#-----  
# Group all probability plots  
#-----  
  
cat<<END>temp  
        Parameters for PLOTEM  
        *****  
  
START OF PARAMETERS:  
$probout  
2 3  
END  
  
for ((x=1 ; x<=N ; x++)); do  
    echo "${x}.ps" >>temp  
done  
  
    plotem temp; rm temp ?.ps  
  
echo "Sim1_EDA.bsh finished"  
#-----  
# Script complete  
#-----  
  
#*****
```

```

#!/bin/bash
#=====
# Script Sim2.3_Dataprep.bsh
#=====
# Prepare data - NS transform and variography
#=====

#read Master_Parameter List

echo "Sim2_Dataprep.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
#source the parameter file
. PPMT_Master_Parameters.txt

infile=$input_data
nvar=$nvar
var1=$var1
var2=$var2
var3=$var3
var4=$var4
var5=$var5
let Wtcol=$((var5+1))

#rock model
rock=$rock

#grid definition
grid=$grid_definition_file

#-----
# Calculate azimuth and dip
#-----

date

echo calculating Principal Azimuth and Dip Directions from Rotation Convention
cat<<END>temp
        Parameters for AZMDIP
        *****

START OF PARAMETERS:
  ${search[0]} ${search[1]} ${search[2]}           -ang1, ang2, ang3
END

echo temp| azmdip |tee azm_dip

```

```

# get azimuth and dip for each axis
awk '/hmax/{print $6, $7}' azm_dip > work/hmax
awk '/hmin/{print $6, $7}' azm_dip > work/hmin
awk '/vert/{print $6, $7}' azm_dip > work/vert

echo "Principal Axis hmax, azimuth and dip : $(cat work/hmax)"
echo "Principal Axis hmin, azimuth and dip : $(cat work/hmin)"
echo "Principal Axis vert, azimuth and dip : $(cat work/vert)"
echo
echo

#write these variables to file data - then load as script variables
awk '{print $1}' work/hmax > work/azm1; az1=$(cat work/azm1)
awk '{print $2}' work/hmax > work/dip1; dip1=$(cat work/dip1)
awk '{print $1}' work/hmin > work/azm2; az2=$(cat work/azm2)
awk '{print $2}' work/hmin > work/dip2; dip2=$(cat work/dip2)
awk '{print $1}' work/vert > work/azm3; az3=$(cat work/azm3)
awk '{print $2}' work/vert > work/dip3; dip3=$(cat work/dip3)

rm work/hmax; rm work/hmin; rm work/vert; rm azm_dip

#-----
# Run casebuild for data summary
#-----

echo run Casebuild program to define the data columns etc...
cat<<END>temp
          Parameters for CASEBUILD
          *****

START OF PARAMETERS:
$infile           -file with data
$nvar  $var1 $var2 $var3 $var4  $var5  -number of variables and columns
-98. 999.         -trimming limits
work/settings.sh  -file for output case statement
END

echo temp |casebuild
dos2unix work/settings.sh

```

```

#-----
# Nearest neighbour decluster
#-----

cat<<END>>temp

                Parameters for DECLUS_NN
                *****

START OF PARAMETERS:
$infile          -input data file
0 1 2 3 $var1     -columns for dh, x, y, z, and the variable...
-0.01  1.0e21    -trimming limits
END

cat rock/$grid > temp0
grep -v Semi temp0>>temp

cat<<END>>>temp
300.0  300.0  30.0          -maximum search radii (hmax,hmin,vert)...
  ${search[0]} ${search[1]} ${search[2]} -angles for search ellipsoid      ...
0                                             -constrain the declustering weights ...
0.01  0.99                    -min. and max. quantiles for well/dh ...
0                                             -constrain the declustering weights ...
0.05  0.95                    -min. and max. quantiles for global ...
1                                             -input keyout indicator grid?      ...
rock/$rock                               -input file with keyout indicator grid...
1 $keep_code                             -column with keyout indicator      ...
work/declus_nn.out                       -output file with declustering weights...
0                                             -output NN model grid?            ...
work/nearestneighbour.gsb                -output file with NN model grid    ...
END

echo temp | declus_nn

#Read the number of columns in declus_nn out file, weight will be the last:
awk 'FNR==2{print $1}' work/declus_nn.out | tr '\n' ' ' > work/Wtcol

Weight=${(<work/Wtcol)}

echo  Decluster Weights are located in Column $Weight of file: declus_nn.out

```

```

#-----
# Report statistics of NN decus data in histograms
#-----
runHist()
{
cat<<END>temp

                Parameters for HISTPLT
                *****

START OF PARAMETERS:
work/declus_nn.out           -file with data
${var}   $Weight             -columns for variable and weight
-1.0     1.0e21              -trimming limits
${num}.ps                     -file for PostScript output
${min}   ${max}              -attribute minimum and maximum
-1.0                                           -frequency maximum (<0 for automatic)
50                                           -number of classes
0                                           -0=arithmetic, 1=log scaling
0                                           -0=frequency, 1=cumulative histogram
0                                           -number of cum. quantiles (<0 for all)
2                                           -number of decimal places (<0 for auto.)
${title}                       -title
1.5                             -positioning of stats (L to R: -1 to 1)
-1.1e21                         -reference value for box plot

END
echo temp | histplt ; rm temp
}

#now loop over variables in file: assemble ps files for each variable
num=0
for ((x=1 ;x<=$nvar ;x++)); do
    . work/settings.sh ${x}
    var=${col}
    minval=${min}
    maxval=${max}
    title="${name} NN Decluster"
    let num=num+1
runHist
done
#-----
# Group all histograms
#-----
cat<<END>temp

                Parameters for PLOTEM
                *****

START OF PARAMETERS:
eda/NN_Declus.ps             -output file
2 3                          -number of plots in X and Y
1.ps                         -first plot file
2.ps                         -second plot file
3.ps                         -third plot file
4.ps                         -fourth plot file
5.ps
nofile
END
echo temp | plotem; rm temp
rm 1.ps; rm 2.ps; rm 3.ps; rm 4.ps; rm 5.ps

```

```

#-----
# Normal Scores of all Variables
#-----

cat<<END>temp

                Parameters for UNSCORE
                *****

START OF PARAMETERS:
work/declus_nn.out      -file with data
$nvar  $var1 $var2 $var3 $var4  $var5  -number of variables and columns
$Weight                -column for weight, 0 if none
0                      -column for category, 0 if none
0                      -number of records if known, 0 if unknown
-0.01  101.0          -trimming limits
0                      -transform using a reference distribution, 1=yes
histsmth.out           -file with reference distribution.
1  2  0               -columns for variable, weight, and category
0                      -maximum number of quantiles, 0 for all
nscore/unscore.out     -file for output
nscore/unscore.trn     -file for output transformation table
END

# run the normal score transform
echo temp |unscore

#remove temp file
rm temp

#-----
# Report statistics of NS data in histograms
#-----

runHist()
{
cat<<END>temp

                Parameters for HISTPLT
                *****

START OF PARAMETERS:
nscore/unscore.out     -file with data
${var}  $Weight        -columns for variable and weight
-998    1.0e21         -trimming limits
${num}.ps              -file for PostScript output
${min}  ${max}         -attribute minimum and maximum
-1.0                    -frequency maximum (<0 for automatic)
50                      -number of classes
0                      -0=arithmetic, 1=log scaling
0                      -0=frequency, 1=cumulative histogram
0                      -number of cum. quantiles (<0 for all)
2                      -number of decimal places (<0 for auto.)
${title}              -title
1.5                    -positioning of stats (L to R: -1 to 1)
-1.1e21               -reference value for box plot

END
echo temp | histplt ; rm temp
}

```

```
#now loop over variables in file: assemble ps files for each variable
```

```
num=0
for ((x=1 ;x<=$nvar ;x++)); do
  . work/settings.sh {x}
  let var={col}+6
  min=-4
  max=4
  title="{name} NS Declust"
  let num=num+1
runHist
done
```

```
#-----
# Group all histograms
#-----
```

```
cat<<END>temp
```

```
Parameters for PLOTTEM
*****
```

```
START OF PARAMETERS:
```

```
nscore/unscore.ps          -output file
2 3                        -number of plots in X and Y
1.ps                       -first plot file
2.ps                       -second plot file
3.ps                       -third plot file
4.ps                       -fourth plot file
5.ps
nofile
END
```

```
echo temp | plotem; rm temp
rm 1.ps; rm 2.ps; rm 3.ps; rm 4.ps; rm 5.ps
```

```
#-----
# Prepare data for NS variography - Re-set var1...varN as the NS transform
#-----
```

```
. work/settings.sh 1
echo "Decluster weight is column {Weight} in file unscore.out"
var1={($Weight+1)}
echo "Variable {name} is column {var1} in file unscore.out"
. work/settings.sh 2
var2={($Weight+2)}
echo "Variable {name} is column {var2} in file unscore.out"
. work/settings.sh 3
var3={($Weight+3)}
echo "Variable {name} is column {var3} in file unscore.out"
. work/settings.sh 4
var4={($Weight+4)}
echo "Variable {name} is column {var4} in file unscore.out"
. work/settings.sh 5
var5={($Weight+5)}
echo "Variable {name} is column {var5} in file unscore.out"
```



```
# Here are some variables that control the variogram postscript files
# Maximum distances for variograms to be plotted dir1, dir2, dir3
maxdis1=200
maxdis2=200
maxdis3=20
```

```
#maximum y value for variogram plots
maxvariog=1.3
```

```
#join experimental variogram points with dashed line 1=yes, 0=no
dash=0
```

```
#plot the variogram sill of 1 (NS Gaussian variable)? 1=yes, 2=no
sill=1
sillval=1.0
```

```
#-----
# Calculate variograms of NS
#-----
```

```
cat<<END>temp
```

```
Parameters for VARCALC
*****
```

```
START OF PARAMETERS:
```

```
nscore/unscore.out      -file with data      ...
$xcol  $ycol  $zcol      -columns for X, Y, Z coordinates  ...
$nvar  $var1  $var2  $var3  $var4  $var5  -number of variables,column numbers (position...
-99.0   99.0             -trimming limits     ...
3                                               -number of directions  ...
$az1 22.5 10 $dip1 22.5 10 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol  ...
 30 25.0 12.5 -number of lags,lag distance,lag tolerance  ...
$az2 22.5 10 $dip2 22.5 10 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol  ...
 30 25.0 12.5 -number of lags,lag distance,lag tolerance  ...
$az3 22.5 100 $dip3 22.5 100 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol  ...
 60 1.0 1 -number of lags,lag distance,lag tolerance  ...
vgram/NSgamv.out      -file for variogram output  ...
0 -legacy output (0=no, 1=write out gamv2004  ...
1 -run checks for common errors  ...
0 -standardize sills? (0=no, 1=yes)  ...
$nvar -number of variogram types  ...
1 1 1 ? -tail variable, head variable, variogram type...
2 2 1 ? -tail variable, head variable, variogram type...
3 3 1 ? -tail variable, head variable, variogram type...
4 4 1 ? -tail variable, head variable, variogram type...
5 5 1 ? -tail variable, head variable, variogram type...
```

```
END
```

```
echo temp | varcalc
cp temp work/varcalc.par
```

```

#-----
# Fit Variogram variable 1
#-----

. work/settings.sh 1

cat<<END>temp

          Parameters for VARMODEL
          *****

START OF PARAMETERS:
vgram/varmodell.out      -file for modeled variogram points output ...
3                        -number of directions to model points along...
$az1  $dip1  1000  1     -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1     -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1   -azm, dip, npoints, point separation ...
2      0.10:0.11        -nst, nugget effect ...
1      0.75  $az1  $dip1  0.0 -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
        25      25      4.5   -a_hmax, a_hmin, a_vert (ranges) ...
1      0.15  $az1  $dip1  0.0 -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
        600     750     5     -a_hmax, a_hmin, a_vert (ranges) ...
1      100000          -fit model (0=no, 1=yes), maximum iteration...
$sillval                -variogram sill (can be fit, but not ...
1                        -number of experimental files to use ...
vgram/NSgamv.out       -experimental output file 1 ...
3 1 $(1+$nvar) $(1+$nvar) -# of variograms (<=0 for all), variogram ...
1 1 10                 -# pairs weighting, inverse distance ...
0      10.0           -fix Hmax/Vert anis. (0=no, 1=yes) ...
0      1.0            -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit${name}.var -file to save fit variogram model ...
END

#NOTES ON VARIOGRAM FITTING:
#1) This program can be run as the GSLIB program vmodel where an already
#    fit variogram model is provided.
#2) Alternatively, a variogram model can be fit. Any parameter, except
#    the number of structures can be fit. Fitting variogram angles
#    is NOT recommended best practice. Options for fitting are:
#    ? - fit the parameter with no constraints
#    a:b - fit the parameter between a and b
#    a: - fit the parameter so it is >=a
#    :b - fit the parameter so it is <=b
#    There must be no spaces in a:b!
#3) Structure types (it) are:
#    1 - spherical variogram model
#    2 - exponential variogram model
#    3 - gaussian variogram model
#    4 - hole effect variogram model (cannot be automatically fit)

echo temp | varmodel

```

```

#-----
# Plot Var1 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
1 $dash 2 $dash 10       -variogram #, dash #, pts?, line?, color ...
vgram/varmodell.out       -2 file with variogram data       ...
1 0 0 1 1               -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot V1 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
${(1+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodell.out       -2 file with variogram data       ...
2 0 0 1 1               -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot V1 Dir 3
#-----
cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
vgram/v3.ps          -file for PostScript output          ...
2                   -number of variograms to plot          ...
0.0 $maxdis3        -distance limits (from data if max<min) ...
0.0 $maxvariog      -variogram limits (from data if max<min) ...
$sill 1.0           -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 3    -Title for variogram                  ...
vgram/NSgamv.out    -1 file with variogram data           ...
$((1+$nvar+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodell1.out -2 file with variogram data           ...
3 0 0 1 1          -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

    Parameters for PLOTTEM
    *****

START OF PARAMETERS:
vgram/NS_${name}_variogram.ps -output file
1 3                             -number of plots in X and Y
vgram/v1.ps                     -first plot file
vgram/v2.ps                     -second plot file
vgram/v3.ps                     -third plot file
END

echo temp | plotem; rm temp
rm ppmt/v1.ps ppmt/v2.ps ppmt/v3.ps

```

```

#-----
# Fit Variogram variable 2
#-----

. work/settings.sh 2

cat<<END>temp
      Parameters for VARMODEL
      *****

START OF PARAMETERS:
vgram/varmodel2.out      -file for modeled variogram points output ...
3                        -number of directions to model points along...
$az1  $dip1  1000  1     -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1     -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1   -azm, dip, npoints, point separation ...
2      0.05:0.06        -nst, nugget effect ...
1      0.9  $az1  $dip1  0.0 -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      10   25   3.4     -a_hmax, a_hmin, a_vert (ranges) ...
1      0.05  $az1  $dip1  0.0 -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      400  350  3.5     -a_hmax, a_hmin, a_vert (ranges) ...
1      100000          -fit model (0=no, 1=yes), maximum iteration...
$sillval                 -variogram sill (can be fit, but not ...
1                        -number of experimental files to use ...
vgram/NSgamv.out        -experimental output file 1 ...
3 2 $((2+$nvar)) $((2+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10                  -# pairs weighting, inverse distance ...
0      10.0             -fix Hmax/Vert anis. (0=no, 1=yes) ...
0      1.0              -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit${name}.var -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot V2 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
2 $dash 2 $dash 10        -variogram #, dash #, pts?, line?, color ...
vgram/varmodel2.out       -2 file with variogram data       ...
1 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot V2 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
${(2+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel2.out       -2 file with variogram data       ...
2 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot V2 Dir 3
#-----
cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
vgram/v3.ps          -file for PostScript output          ...
2                   -number of variograms to plot      ...
0.0 $maxdis3        -distance limits (from data if max<min) ...
0.0 $maxvariog      -variogram limits (from data if max<min) ...
$sill 1.0           -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 3    -Title for variogram                  ...
vgram/NSgamv.out    -1 file with variogram data            ...
$((2+$nvar+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel2.out -2 file with variogram data            ...
3 0 0 1 1          -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

    Parameters for PLOTTEM
    *****

START OF PARAMETERS:
vgram/NS_${name}_variogram.ps -output file
1 3                            -number of plots in X and Y
vgram/v1.ps                    -first plot file
vgram/v2.ps                    -second plot file
vgram/v3.ps                    -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Var 3
#-----

. work/settings.sh 3

cat<<END>temp
      Parameters for VARMODEL
      *****

START OF PARAMETERS:
vgram/varmodel3.out      -file for modeled variogram points output ...
3      -number of directions to model points along...
$az1  $dip1  1000  1      -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1      -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1    -azm, dip, npoints, point separation ...
2      0.1:0.11          -nst, nugget effect ...
1      0.60  $az1  $dip1  0.0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      15      20      5      -a_hmax, a_hmin, a_vert (ranges) ...
1      0.30  $az1  $dip1  0.0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      250     600    6      -a_hmax, a_hmin, a_vert (ranges) ...
1      100000          -fit model (0=no, 1=yes), maximum iteration...
$sillval                -variogram sill (can be fit, but not ...
1                        -number of experimental files to use ...
vgram/NSgamv.out        -experimental output file 1 ...
3 3 $((3+$nvar)) $((3+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10                  -# pairs weighting, inverse distance ...
0      10.0             -fix Hmax/Vert anis. (0=no, 1=yes) ...
0      1.0              -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit${name}.var -file to save fit variogram model ...
END

echo temp | varmodel

```



```

#-----
# Plot V3 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $maxvariog           -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
3 $dash 2 $dash 10        -variogram #, dash #, pts?, line?, color ...
vgram/varmodel3.out       -2 file with variogram data       ...
1 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot V3 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $maxvariog           -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
${(3+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel3.out       -2 file with variogram data       ...
2 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot V3 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output          ...
2                          -number of variograms to plot        ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram                  ...
vgram/NSgamv.out          -1 file with variogram data           ...
$(3+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel3.out       -2 file with variogram data           ...
3 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
vgram/NS_${name}_variogram.ps -output file
1 3                             -number of plots in X and Y
vgram/v1.ps                     -first plot file
vgram/v2.ps                     -second plot file
vgram/v3.ps                     -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Var 4
#-----

. work/settings.sh 4
cat<<END>temp

          Parameters for VARMODEL
          *****

START OF PARAMETERS:
vgram/varmodel4.out      -file for modeled variogram points output ...
3                        -number of directions to model points along...
$az1  $dip1  1000  1     -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1     -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1   -azm, dip, npoints, point separation ...
2      0.05:0.06        -nst, nugget effect ...
1      0.75  $az1  $dip1  0.0   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
          12    12    3.7     -a_hmax, a_hmin, a_vert (ranges) ...
1      0.20  $az1  $dip1  0.0   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
          400  470  4         -a_hmax, a_hmin, a_vert (ranges) ...
1      100000          -fit model (0=no, 1=yes), maximum iteration...
$sillval                -variogram sill (can be fit, but not ...
1                        -number of experimental files to use ...
vgram/NSgamv.out        -experimental output file 1 ...
3 4 $((4+$nvar)) $((4+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10                  -# pairs weighting, inverse distance ...
0      10.0             -fix Hmax/Vert anis. (0=no, 1=yes) ...
0      1.0              -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit${name}.var -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot V4 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram            ...
vgram/NSgamv.out          -1 file with variogram data      ...
4 $dash 2 $dash 10        -variogram #, dash #, pts?, line?, color ...
vgram/varmodel4.out       -2 file with variogram data      ...
1 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot V4 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram            ...
vgram/NSgamv.out          -1 file with variogram data      ...
${(4+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel4.out       -2 file with variogram data      ...
2 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot V4 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output          ...
2                          -number of variograms to plot        ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill      1.0            -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram                  ...
vgram/NSgamv.out          -1 file with variogram data           ...
$(4+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel4.out      -2 file with variogram data           ...
3 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
vgram/NS_${name}_variogram.ps -output file
1 3                            -number of plots in X and Y
vgram/v1.ps                    -first plot file
vgram/v2.ps                    -second plot file
vgram/v3.ps                    -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Var 5
#-----

. work/settings.sh 5

cat<<END>temp
      Parameters for VARMODEL
      *****

START OF PARAMETERS:
vgram/varmodel5.out          -file for modeled variogram points output ...
3                            -number of directions to model points along...
$az1  $dip1  1000  1         -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1         -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1       -azm, dip, npoints, point separation ...
2    0.05:0.06              -nst, nugget effect ...
1    0.60  $az1  $dip1  0.0   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      20    15    8          -a_hmax, a_hmin, a_vert (ranges) ...
1    0.35  $az1  $dip1  0.0   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      600   600   12        -a_hmax, a_hmin, a_vert (ranges) ...
1    100000                 -fit model (0=no, 1=yes), maximum iteration...
$sillval                     -variogram sill (can be fit, but not ...
1                             -number of experimental files to use ...
vgram/NSgamv.out             -experimental output file 1 ...
3 5 $((5+$nvar)) $((5+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10                       -# pairs weighting, inverse distance ...
0    10.0                    -fix Hmax/Vert anis. (0=no, 1=yes) ...
0    1.0                     -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit${name}.var     -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot V5 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill      1.0            -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
5  $dash 2  $dash 10     -variogram #, dash #, pts?, line?, color ...
vgram/varmodel5.out      -2 file with variogram data       ...
1  0  0  1  1           -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot V5 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill      1.0            -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram              ...
vgram/NSgamv.out          -1 file with variogram data       ...
${(5+$nvar)} $dash 2  $dash 10     -variogram #, dash #, pts?, line?, color ...
vgram/varmodel5.out      -2 file with variogram data       ...
2  0  0  1  1           -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot V5 Dir 3
#-----

cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
vgram/v3.ps          -file for PostScript output          ...
2                   -number of variograms to plot      ...
0.0 $maxdis3        -distance limits (from data if max<min) ...
0.0                 -variogram limits (from data if max<min) ...
$maxvariog          -plot sill (0=no,1=yes,2+=multiple), sill...
$sill 1.0           -Title for variogram                  ...
NS ${name} Dir 3    -1 file with variogram data            ...
vgram/NSgamv.out    -variogram #, dash #, pts?, line?, color ...
$(5+$nvar+$nvar) $dash 2 $dash 10    -2 file with variogram data            ...
vgram/varmodel5.out -variogram #, dash #, pts?, line?, color ...
3 0 0 1 1

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

    Parameters for PLOTTEM
    *****

START OF PARAMETERS:
vgram/NS_${name}_variogram.ps -output file
1 3                             -number of plots in X and Y
vgram/v1.ps                     -first plot file
vgram/v2.ps                     -second plot file
vgram/v3.ps                     -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

#-----
# Done with Normal Score Variography
#-----

```



```

#-----
# Calculate and Model Raw Data Variograms/Inverse Correlograms: refresh the variable list
#-----

. PPMT_Master_Parameters.txt
infile=$input_data
nvar=$nvar
var1=$var1
var2=$var2
var3=$var3
var4=$var4
var5=$var5

echo Calculating and modelling Raw Data Variograms/Inverse Correlograms

cat<<END>temp

                Parameters for VARCALC
                *****

START OF PARAMETERS:
$infile          -file with data          ...
$xcoll $ycoll $zcoll -columns for X, Y, Z coordinates      ...
$nvar $var1 $var2 $var3 $var4 $var5 -number of variables,column numbers (position...
-0.001 101.0     -trimming limits          ...
3               -number of directions        ...
$az1 22.5 10 $dip1 22.5 10 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol ...
 30 25.0 12.5    -number of lags,lag distance,lag tolerance ...
$az2 22.5 10 $dip2 22.5 10 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol ...
 30 25.0 12.5    -number of lags,lag distance,lag tolerance ...
$az3 22.5 100 $dip3 22.5 100 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol ...
 60 1.0 1        -number of lags,lag distance,lag tolerance ...
vgram/Rawgamv.out -file for variogram output            ...
0                -legacy output (0=no, 1=write out gamv2004 ...
1                -run checks for common errors          ...
0                -standardize sills? (0=no, 1=yes)      ...
$nvar            -number of variogram types            ...
1 1 1 ?         -tail variable, head variable, variogram type...
2 2 1 ?         -tail variable, head variable, variogram type...
3 3 1 ?         -tail variable, head variable, variogram type...
4 4 1 ?         -tail variable, head variable, variogram type...
5 5 1 ?         -tail variable, head variable, variogram type...

END

echo temp | varcalc
cp temp work/varcalc.par

```

```

#-----
# Fit Raw Data Variogram Var 1
#-----

#echo "${sdev}*${sdev}" | bc > work/variance
#sillval=$(<work/variance)

#Use this option for declustered variance
#-----
#Create files with the declustered variance of each variable
awk 'c&&!-c;/std. dev./{c=2}' eda/NN_Declus.ps | sed 's/[^0-9.]*/g' > work/dc_sds

#write declustered variances into the work folder
for ((x=1 ; x<=$nvar ; x++)); do
    . work/settings.sh ${x}
    awk -v vname=${name} -v rec=${x} '{if(NR==rec) print $1*$1}' work/dc_sds >
    work/dc_variance_${name}
done
#-----

    . work/settings.sh 1

sillval=$(<work/dc_variance_NI)

echo "${sillval}*1.3" | bc > work/xlim
xlim=$(<work/xlim)

cat<<END>temp
                Parameters for VARMODEL
                *****

START OF PARAMETERS:
vgram/varmodel${name}_declust_var.out    -file for modeled variogram points output ...
3                                          -number of directions to model points along...
$az1  $dip1  1000  1                      -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1                      -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1                    -azm, dip, npoints, point separation ...
  2    0.00957:0.00958                    -nst, nugget effect ...
  1    0.14361      0.00000                -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      4.00000      4.00000                -a_hmax, a_hmin, a_vert (ranges) ...
  1    0.01214      0.00000                -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      10.00000     15.00000                -a_hmax, a_hmin, a_vert (ranges) ...
1  100000                                -fit model (0=no, 1=yes), maximum iteration...
$sillval                                -variogram sill (can be fit, but not ...
1                                          -number of experimental files to use ...
vgram/Rawgamv.out                        -experimental output file 1 ...
3 1 $((1+$nvar)) $((1+$nvar+$nvar))      -# of variograms (<=0 for all), variogram ...
1  1  10                                  -# pairs weighting, inverse distance ...
0  10.0                                    -fix Hmax/Vert anis. (0=no, 1=yes) ...
0  1.0                                      -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit_raw${name}_declust_var.var  -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Raw Var1 Dir 1
#-----

cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram            ...
vgram/Rawgamv.out          -1 file with variogram data      ...
1 $dash 2 $dash 10        -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
1 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Raw Var1 Dir 2
#-----

cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram            ...
vgram/Rawgamv.out          -1 file with variogram data      ...
${(1+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
2 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Raw Var1 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output          ...
2                          -number of variograms to plot        ...
0.0 $maxdis3               -distance limits (from data if max<min) ...
0.0 $xlim                  -variogram limits (from data if max<min) ...
$sill $sillval             -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram                  ...
vgram/Rawgamv.out          -1 file with variogram data           ...
$(1+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data           ...
3 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
vgram/Raw_${name}_variogram_declust_var.ps -output file
1 3 -number of plots in X and Y
vgram/v1.ps -first plot file
vgram/v2.ps -second plot file
vgram/v3.ps -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Raw Var 2
#-----

. work/settings.sh 2

#echo "${sdev}*${sdev}" | bc > work/variance
#sillval=$(work/variance)

#Use this option for declustered variance
sillval=$(work/dc_variance_SIO2)

echo "${sillval}*1.3" | bc > work/xlim
xlim=$(work/xlim)

cat<<END>temp
                Parameters for VARMODEL
                *****

START OF PARAMETERS:
vgram/varmodel${name}_declust_var.out  -file for modeled variogram points output ...
3                                         -number of directions to model points along...
$az1  $dip1  1000  1                       -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1                       -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1                     -azm, dip, npoints, point separation ...
2                                           -nst, nugget effect ...
1      26.31311  0.000  0.0  0.0          -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      25.00000  25.00  4.0                -a_hmax, a_hmin, a_vert (ranges) ...
1      4.46520  0.000  0.0  0.0          -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
      500.00000  300.0  5.0                -a_hmax, a_hmin, a_vert (ranges) ...
1  100000                                     -fit model (0=no, 1=yes), maximum iteration...
$sillval                                       -variogram sill (can be fit, but not ...
1                                             -number of experimental files to use ...
vgram/Rawgamv.out                             -experimental output file 1 ...
3 2 $(2+$nvar) $(2+$nvar+$nvar)              -# of variograms (<=0 for all), variogram ...
1 1 10                                         -# pairs weighting, inverse distance ...
0 10.0                                         -fix Hmax/Vert anis. (0=no, 1=yes) ...
0 1.0                                           -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit_raw${name}_declust_var.var      -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Raw V2 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram            ...
vgram/Rawgamv.out         -1 file with variogram data      ...
2 $dash 2 $dash 10       -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
1 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Raw V2 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram            ...
vgram/Rawgamv.out         -1 file with variogram data      ...
${(2+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
2 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Raw V2 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram              ...
vgram/Rawgamv.out         -1 file with variogram data       ...
${((2+$nvar+$nvar)) $dash 2 $dash 10} -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data       ...
3 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
vgram/Raw_${name}_variogram_declust_var.ps -output file
1 3                                         -number of plots in X and Y
vgram/v1.ps                                -first plot file
vgram/v2.ps                                -second plot file
vgram/v3.ps                                -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Raw Var 3
#-----

. work/settings.sh 3

#echo "${sdev}*${sdev}" | bc > work/variance
#sillval=$(

```



```

#-----
# Plot Raw V3 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram              ...
vgram/Rawgamv.out         -1 file with variogram data       ...
3 $dash 2 $dash 10       -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data       ...
1 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Raw V3 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
NS ${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram              ...
vgram/Rawgamv.out         -1 file with variogram data       ...
${(3+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data       ...
2 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Raw V3 Dir 3
#-----

cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output          ...
2                          -number of variograms to plot        ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram                  ...
vgram/Rawgamv.out         -1 file with variogram data           ...
$( (3+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data           ...
3 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

    Parameters for PLOTTEM
    *****

START OF PARAMETERS:
vgram/Raw_${name}_variogram_declust_var.ps -output file
1 3 -number of plots in X and Y
vgram/v1.ps -first plot file
vgram/v2.ps -second plot file
vgram/v3.ps -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Raw V4
#-----

. work/settings.sh 4

#echo "${sdev}*${sdev}" | bc > work/variance
#sillval=$(

```

```

#-----
# Plot Raw V4 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram            ...
vgram/Rawgamv.out         -1 file with variogram data      ...
4 $dash 2 $dash 10       -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
1 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Raw V4 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram            ...
vgram/Rawgamv.out         -1 file with variogram data      ...
${(4+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
2 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Raw V4 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram            ...
vgram/Rawgamv.out         -1 file with variogram data      ...
$(4+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data      ...
3 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
vgram/Raw_${name}_variogram_declust_var.ps -output file
1 3 -number of plots in X and Y
vgram/v1.ps -first plot file
vgram/v2.ps -second plot file
vgram/v3.ps -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps

```

```

#-----
# Fit Variogram Raw Var 5
#-----

. work/settings.sh 5

#echo "${sdev}*${sdev}" | bc > work/variance
#sillval=$(<work/variance)

#Use this option for declustered variance
sillval=$(<work/dc_variance_CAO)

echo "${sillval}*1.3" | bc > work/xlim
xlim=$(<work/xlim)

cat<<END>temp
                Parameters for VARMODEL
                *****

START OF PARAMETERS:
vgram/varmodel${name}_declust_var.out  -file for modeled variogram points output ...
3                                         -number of directions to model points along...
$az1  $dip1  1000  1                       -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1                       -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1                     -azm, dip, npoints, point separation ...
2  0.00497:0.005                          -nst, nugget effect ...
1  0.06457 0.000 0.0 0.0                   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    20.00000 20.00 7.0                     -a_hmax, a_hmin, a_vert (ranges) ...
1  0.04566 0.000 0.0 0.0                   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    450.00000 400.0 9.0                     -a_hmax, a_hmin, a_vert (ranges) ...
1  100000                                     -fit model (0=no, 1=yes), maximum iteration...
$sillval                                     -variogram sill (can be fit, but not ...
1                                             -number of experimental files to use ...
vgram/Rawgamv.out                             -experimental output file 1 ...
3 5 $(5+$nvar) $(5+$nvar+$nvar)             -# of variograms (<=0 for all), variogram ...
1 1 10                                         -# pairs weighting, inverse distance ...
0 10.0                                         -fix Hmax/Vert anis. (0=no, 1=yes) ...
0 1.0                                           -fix Hmin/Hmax anis. (0=no, 1=yes) ...
vgram/varfit_raw${name}_declust_var.var     -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Raw V5 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 1 azm:$az1 dip:$dip1 -Title for variogram              ...
vgram/Rawgamv.out         -1 file with variogram data       ...
5 $dash 2 $dash 10       -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data       ...
1 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Raw V5 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $xlim                 -variogram limits (from data if max<min) ...
$sill $sillval            -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 2 azm:$az2 dip:$dip2 -Title for variogram              ...
vgram/Rawgamv.out         -1 file with variogram data       ...
${(5+$nvar)} $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data       ...
2 0 0 1 1                -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Raw V5 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
vgram/v3.ps                -file for PostScript output          ...
2                          -number of variograms to plot        ...
0.0 $maxdis3               -distance limits (from data if max<min) ...
0.0 $xlim                  -variogram limits (from data if max<min) ...
$sill $sillval             -plot sill (0=no,1=yes,2+=multiple), sill...
${name} Dir 3 azm:$az3 dip:$dip3 -Title for variogram                  ...
vgram/Rawgamv.out          -1 file with variogram data           ...
$(5+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
vgram/varmodel${name}_declust_var.out -2 file with variogram data           ...
3 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
vgram/Raw_${name}_variogram_declust_var.ps -output file
1 3 -number of plots in X and Y
vgram/v1.ps -first plot file
vgram/v2.ps -second plot file
vgram/v3.ps -third plot file
END

echo temp | plotem; rm temp
rm vgram/v1.ps vgram/v2.ps vgram/v3.ps
rm temp0

date
echo "Sim2_Dataprep.bsh finished"
#=====
# Script complete
#=====

#*****

```



```

#!/bin/bash
#=====
# Script Sim3.2_Impute.bsh
#=====
# Impute data and PPMT transform
#=====

#read Master_Parameter List

echo "Sim3.2_Impute.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt

infile=$input_data
nvar=$nvar
var1=$var1
var2=$var2
var3=$var3
var4=$var4
var5=$var5

#rockmodel
rock=$rock

#grid_definition
grid=$grid_definition_file

#how many histogram classes for PPMT Histograms
nclass=50

```

```

#-----
# Principal directions for variogram calculation-read from files -load with error checks
#-----
if [ -e work/azm1 ] ; then
az1=$(cat work/azm1)
echo "reading variogram orientation data"
else
echo "Fatal Error: azm1 file not found"
exit 1
fi
if [ -e work/dip1 ] ; then
dip1=$(cat work/dip1)
else
echo "Fatal Error: dip1 file not found"
fi
if [ -e work/azm2 ] ; then
az2=$(cat work/azm2)
else
echo "Fatal Error: azm2 file not found"
exit 1
fi
if [ -e work/dip2 ] ; then
dip2=$(cat work/dip2)
else
echo "Fatal Error: dip2 file not found"
exit 1
fi
if [ -e work/azm3 ] ; then
az3=$(cat work/azm3)
else
echo "Fatal Error: azm3 file not found"
exit 1
fi
if [ -e work/dip3 ] ; then
dip3=$(cat work/dip3)
else
echo "Fatal Error: dip3 file not found"
exit 1
fi
date

#-----
# Casebuild to re-create settings file controlling this run
#-----

cat<<END>temp
          Parameters for CASEBUILD
          *****

START OF PARAMETERS:
$infile                                     -file with data
$nvar  $var1 $var2 $var3 $var4  $var5      -number of variables and columns
-98. 999.                                   -trimming limits
work/settings.sh                            -file for output case statement
END

echo temp |casebuild
dos2unix work/settings.sh

```

```

#-----
# Data imputation to fill in missing values
#-----
rm temp0
for ((x=1 ; x<=$nvar ; x++)); do
    . work/settings.sh ${x}
file="vgram/varfit${name}.var"
echo processing $file
dos2unix $file
cat $file >> temp0
done
grep -v Semi temp0 > temp1 ; rm temp0
#read the ID of the DC weight column
Weight=$(work/Wtcol)
    . work/settings.sh 1
echo "Decluster weight is column ${Weight} in file unscore.out"
v1=$((Weight+1))
echo "Variable NS ${name} is column ${v1} in file unscore.out"
    . work/settings.sh 2
v2=$((Weight+2))
echo "Variable NS ${name} is column ${v2} in file unscore.out"
    . work/settings.sh 3
v3=$((Weight+3))
echo "Variable NS ${name} is column ${v3} in file unscore.out"
    . work/settings.sh 4
v4=$((Weight+4))
echo "Variable NS ${name} is column ${v4} in file unscore.out"
    . work/settings.sh 5
v5=$((Weight+5))
echo "Variable NS ${name} is column ${v5} in file unscore.out"

cat<<END>temp
                Parameters for IMPUTE
                *****

START OF PARAMETERS:
DATA:
nscore/unscore.out          -file with data to impute (variables  ...
1 2 3                       -columns for x y z coordinates        ...
5                           -number of variables to impute        ...
$v1 $v2 $v3 $v4 $v5        -columns for variables to impute      ...
-998  1.0e21                -trimming limits                      ...
OUTPUT:                                                              ...
work/di_info.out           -output file for imputation means and ...
.\Imp_Real                 -output folder for imputation realizati...
.\Data_Real                -output folder for simulated data files...
1                           -number of data fields to output, other...
$Weight                    -additional data field columns        ...
SIMULATION:                                                         ...
10                          -number of realizations                ...
20                          -maximum data/previously simulated node...
${radii[0]} ${radii[1]} ${radii[2]} -maximum search radii (hmax,hmin,vert) ...
${search[0]} ${search[1]} ${search[2]} -angles for search ellipsoid          ...
69696                       -random number seed1                  ...
1                            -heterotopic correlation? (0=no,1=yes) ...
1                            -non-parametric? (0=no,1=yes)         ...
100 0.08                    -# cond. disc. and kernel bandwidth (if...
10                           -# Gibbs iterations before extraction ...
VARIOGRAMS:
END

```

```

#append variograms to the parameter file
cat templ >> temp
cp temp ppmt/audit_impute_par
echo temp | impute
rm temp templ
#-----
# Back Transform the NS Imputed Data prior to PPMT...normal scores back transform
#-----
dos2unix work/settings.sh
echo Locating Min and Max Values for NScore backtransform
# put mins and max values into variable files
for ((x=1 ;x<=$nvar ;x++));do
. work/settings.sh ${x}
echo ${min} > work/tempmin${x}
echo ${max} >> work/tempmin${x}
cat work/tempmin${x} | tr '\n' ' ' > work/limits${x};
echo Minimum and Maximum Values for ${name} = $(<work/limits${x})
done
rm work/tempmin*
#-----
# Back-transform all the imputed datasets to real space
#-----
runNsb()
{
for ((i=1 ; i<11 ; i++)); do
cat<<END>temp
                Parameters for NSCOREMV_B
                *****

START OF PARAMETERS:
./Data_Real/${i}.dat          -file with data
5                             -# of normal var. to back transform
5 6 7 8 9                    -column #s with normal variables
-998 999                      -trimming limits
./Data_Real/BT_ImpData${i}.dat -file for output
nscore/unscore1.trn          -file with input transformation for var1
nscore/unscore2.trn          -file with input transformation for var2
nscore/unscore3.trn          -file with input transformation for var3
nscore/unscore4.trn          -file with input transformation for var4
nscore/unscore5.trn          -file with input transformation for var5
$(<work/limits1)             -minimum and maximum data value for var1
1 0.0                        -lower tail option and parameter for var1
1 2.0                        -upper tail option and parameter for var1
$(<work/limits2)             -minimum and maximum data value for var2
1 0.0                        -lower tail option and parameter for var2
1 2.0                        -upper tail option and parameter for var2
$(<work/limits3)             -minimum and maximum data value for var3
1 0.0                        -lower tail option and parameter for var3
1 2.0                        -upper tail option and parameter for var3
$(<work/limits4)             -minimum and maximum data value for var4
1 0.0                        -lower tail option and parameter for var4
1 2.0                        -upper tail option and parameter for var4
$(<work/limits5)             -minimum and maximum data value for var5
1 0.0                        -lower tail option and parameter for var5
1 2.0                        -upper tail option and parameter for var5
END
echo temp | nscoremv_b
done
rm tempmin* limits*
date

```

```

#-----
# Casebuild to re-create settings file controlling the check
#-----

cat<<END>temp
        Parameters for CASEBUILD
        *****

START OF PARAMETERS:
./Data_Real/BT_ImpData1.dat           -file with data
$nvar 10 11 12 13 14                 -number of variables and columns
0.000000001 100                       -trimming limits
work/settings.sh                       -file for output case statement
END

echo temp |casebuild
dos2unix work/settings.sh

#-----
# Identify which column is NI and which is the imputed variable for check
#-----

for ((x=1 ; x<=$nvar ; x++)); do

    . work/settings.sh ${x}
    echo $name $col

VFname="MGO" #Variable Focus (VF) - Replace this variable by the desirable one
if [ $name == "Nscore_b:NS_{$VFname}" ]; then
echo $col>VFCol
    echo "${min}*0.9" | bc > work/tempmin
    echo "${max}*1.1" | bc >> work/tempmin
    cat work/tempmin | tr '\n' ' ' > work/vfimp_limits;
    echo Minimum and Maximum Values for ${name} = $(<work/vfimp_limits)
fi

if [ $name == "Nscore_b:NS_NI" ]; then
echo $col>Nicol
    echo "${min}*0.9" | bc > work/tempminni
    echo "${max}*1.1" | bc >> work/tempminni
    cat work/tempminni | tr '\n' ' ' > work/vfimp_limitsni;
    echo Minimum and Maximum Values for ${name} = $(<work/vfimp_limitsni)
fi
done

VFCol=$(cat "VFCol")
Nicol=$(cat "Nicol")

rm work/tempmin
rm work/tempminni

echo
echo
echo Column Number for $VFname in Back Transformed Imputed Data File: VFCol = $VFCol
echo Column Number for Ni in Declustered Data set File: Ni_col= $Nicol
echo

```

```

#-----
# Casebuild to re-create settings file controlling the check
#-----
cat<<END>temp

        Parameters for CASEBUILD
        *****

START OF PARAMETERS:
./work/declus_nn.out          -file with data
$nvar 7 8 9 10 11            -number of variables and columns
0.000000001 100              -trimming limits
work/settings1.sh            -file for output case statement
END

echo temp |casebuild
dos2unix work/settings1.sh

#-----
# Identify which column is the desired imputed variable
#-----
for ((x=1 ; x<=$nvar ; x++)); do

    . work/settings1.sh ${x}
    echo $name $col

if [ $name == ${VFname} ]; then
    echo $col>VFColl
fi
done

VFColl=$(cat "VFColl")

#-----
# Check the Imputed Data sets...focus on Imputed Variable
#-----
cat<<END>temp

        Parameters for HISTPLT
        *****

START OF PARAMETERS:
./work/declus_nn.out          -file with data
$VFColl 12                    -columns for variable and weight
0.000000001 100              -trimming limits
101.ps                        -file for PostScript output
$(<work/vfimp_limits)         -attribute minimum and maximum
-1.0                          -frequency maximum (<0 for automatic)
$class                        -number of classes
0                              -0=arithmetic, 1=log scaling
1                              -0=frequency, 1=cumulative histogram
0                              -number of cum. quantiles (<0 for all)
2                              -number of decimal places (<0 for auto.)
${VFname} Raw Data            -title
1.5                           -positioning of stats (L to R: -1 to 1)
-1.1e21                       -reference value for box plot
END

echo temp | histplt; rm temp

```

```

#-----
# Loop over all the BT imputed data sets
#-----
for ((i=1 ; i<11 ; i++)); do
cat<<END>temp

        Parameters for HISTPLT
        *****

START OF PARAMETERS:
./Data_Real/BT_ImpData${i}.dat      -file with data
$VFCol      4                       -columns for variable and weight
0.000000001  100                    -trimming limits
${i}.ps                                       -file for PostScript output
$(<work/vfimp_limits)                 -attribute minimum and maximum
-1.0                                     -frequency maximum (<0 for automatic)
$nclass                                     -number of classes
0                                         -0=arithmetic, 1=log scaling
1                                         -0=frequency, 1=cumulative histogram
0                                         -number of cum. quantiles (<0 for all)
2                                         -number of decimal places (<0 for auto.)
Imputed ${VFname} ${i}                -title
1.5                                     -positioning of stats (L to R: -1 to 1)
-1.1e21                                  -reference value for box plot
END

echo temp | histplt; rm temp

done

#-----
# Assemble a Plot of these Histograms
#-----
cat<<END>temp

        Parameters for PLOTTEM
        *****

START OF PARAMETERS:
eda/RawSpace_Imputed_${VFname}.ps     -output file
3 4                                     -number of plots in X and Y
101.ps                                  -first plot file
1.ps                                    -second plot file
2.ps                                    - ...
3.ps                                    - ...
4.ps                                    - ...
5.ps
6.ps
7.ps
8.ps
9.ps
10.ps
nofile
nofile
END

echo temp | plotem; rm temp

rm [0-9]*.ps
rm [0-9].ps

```

```

#-----
# Scatterplots of the Normal Score Data Imputed Variable vs Ni
#-----
cat<<END>temp

                Parameters for SCATPLT
                *****

START OF PARAMETERS:
1                -number of data files to plot                ...
./work/declus_nn.out -main data file (will calculate statistics, ...
7  $VFColl  0  0    -columns for X, Y, wt, third var.                ...
0.000000001  100   -trimming limits                ...
0  0  2  1  -1     -line width (0=none), dashing, bullet size ...
101.ps         -file for Postscript output                ...
$(<work/vfimp_limitsni)  0 -X min and max, (0=arith, 1=log) (max<min for...
$(<work/vfimp_limits)  0  -Y min and max, (0=arith, 1=log) (max<min for...
1                -plot every nth data point                ...
0.0            20    -limits for third variable gray scale                ...
1                -reference stats to plot(-1=none,0=cor only, ...
1                -positioning of stats (L to R: -1 to 1)    ...
0                -plot x/y histograms of the first data file ...
Ni vs ${VFname}  -title                ...
default         -X axis label (default=get from data file #1)...
default         -y axis label (default=get from data file #1)...
END

echo temp | scatplt2008; rm temp

#-----
# Get scatterplots from all imputed data sets
#-----
for ((i=1 ; i<=10 ; i++)); do
cat<<END>temp

                Parameters for SCATPLT
                *****

START OF PARAMETERS:
1                -number of data files to plot                ...
./Data_Real/BT_ImpData${i}.dat -main data file (will calculate statistics, ...
$(<NiCol)  $(<VFCol)  0  0    -columns for X, Y, wt, third var.                ...
0.000000001  100   -trimming limits                ...
0  0  2  7  -1     -line width (0=none), dashing, bullet size ...
${i}.ps         -file for Postscript output                ...
$(<work/vfimp_limitsni)  0 -X min and max, (0=arith, 1=log) (max<min for...
$(<work/vfimp_limits)  0  -Y min and max, (0=arith, 1=log) (max<min for...
1                -plot every nth data point                ...
0.0            20    -limits for third variable gray scale                ...
1                -reference stats to plot(-1=none,0=cor only, ...
1                -positioning of stats (L to R: -1 to 1)    ...
0                -plot x/y histograms of the first data file ...
Ni vs Imputed ${VFname} ${i} -title                ...
default         -X axis label (default=get from data file #1)...
default         -y axis label (default=get from data file #1)...
END

echo temp | scatplt2008 ;rm temp
done

```



```

#-----
# Assemble a Single Plot of these Scatterplots
#-----
cat<<END>>temp

                Parameters for PLOTTEM
                *****

START OF PARAMETERS:
eda/RawSpace_Imputed_Ni- $\{VFname\}$ .ps      -output file
3 4                                           -number of plots in X and Y
101.ps                                        -first plot file
1.ps                                         -second plot file
2.ps                                         -third plot file
3.ps                                         -fourth plot file
4.ps
5.ps
6.ps
7.ps
8.ps
9.ps
10.ps
nofile
nofile
END
echo temp | plotem; rm temp
rm 101.ps 1.ps 2.ps 3.ps 4.ps 5.ps 6.ps 7.ps 8.ps 9.ps 10.ps

#-----
# Decluster the [Back-transformed + Imputed] Data Set
#-----
# This step is only needed in case the main variable had some missing values
# Then the weight column number needs changing in the PPMT step
cat<<END>>temp

                Parameters for DECLUS_NN
                *****

START OF PARAMETERS:
Data_Real\BT_ImpData $\{1\}$ .dat      -input data file
0 1 2 3 10                                   -columns for dh, x, y, z, and the varia...
-0.01 1.0e21                                 -trimming limits      ...
END                                           ...
cat rock/$grid > temp0                       ...
grep -v Semi temp0 >> temp                   ...
cat<<END>>>temp                             ...
300.0 300.0 30.0                             -maximum search radii (hmax,hmin,vert) ...
 $\{search[0]\}$   $\{search[1]\}$   $\{search[2]\}$  -angles for search ellipsoid      ...
0                                             -constrain the declustering weights to ...
0.01 0.99                                   -min. and max. quantiles for well/dh ...
0                                             -constrain the declustering weights to ...
0.05 0.95                                   -min. and max. quantiles for global ...
1                                             -input keyout indicator grid?      ...
rock/$rock                                  -input file with keyout indicator grid ...
1 $keep_code                                -column with keyout indicator      ...
work/Imp_declus.out                         -output file with declustering weights ...
0                                             -output NN model grid?             ...
work/nearestneighbour.gsb                  -output file with NN model grid    ...
END

echo temp | declus_nn ; rm temp

```

```

#-----
# Do the PPMT transform...We need the Weight Column from the Imputed Data [wtcol=15]
#-----
# Check column assignments from Imp_declus.out file!!

cat<<END>temp

                                Parameters for PPMT
                                *****

START OF PARAMETERS:
work/Imp_declus.out             -input data file
$Nvar 10 11 12 13 14 4         -number of variables, variable cols, and wt col
-0.01 101.0                    -trimming limits
25 100 50                      -min/max iterations and targeted Gauss perc. (see Note 1)
1                               -spatial decorrelation? (0=no,1=yes) (see Note 2)
1 2 3                           -x, y, z columns (0=none for z)
25 12.5                         -lag distance, lag tolerance
ppmt/ppmt_nscore${1}.out       -output data file with normal score transformed variables
ppmt/ppmt${1}.out              -output data file with PPMT transformed variables
ppmt/ppmt${1}.trn              -output transformation table (binary)
END
echo temp | ppmt ; rm temp
} #end of runNsb
for i in {1..1};do
    runNsb $i
done
date

#-----
# Evaluate PPMT Transform: Histograms
#-----

runHistplt()
{
cat<<END> temp

                                Parameters for HISTPLT
                                *****

START OF PARAMETERS:
ppmt/ppmt${1}.out              -file with data
${2} 4                          -columns for variable and weight
-1.0E21 1.0E21                  -trimming limits
${3}.ps                          -file for PostScript output
-4 4                              -attribute minimum and maximum
-1.0                              -frequency maximum (<0 for automatic)
$Nclass                          -number of classes
0                                 -0=arithmetic, 1=log scaling
0                                 -0=frequency, 1=cumulative histogram
0                                 -number of cum. quantiles (<0 for all)
2                                 -number of decimal places (<0 for auto.)
Var: ${4} PDF                    -title
0.5                              -positioning of stats (L to R: -1 to 1)
-1.0E21                          -reference value for box plot
END
echo temp | histplt; rm temp
}

runHistplt 1 16 1 Ni
runHistplt 1 17 2 SiO
runHistplt 1 18 3 MgO
runHistplt 1 19 4 Fe
runHistplt 1 20 5 CaO

```

```

#-----
# Assemble a Single Plot of these Histograms
#-----

cat<<END>temp
                Parameters for PLOTTEM
                *****

START OF PARAMETERS:
ppmt/PPMT_histograms.ps      -output file
2 3                          -number of plots in X and Y
1.ps
2.ps
3.ps
4.ps
5.ps
END

plotem temp; rm temp ?.ps 1?.ps

#-----
# Evaluate PPMT Transform: Scatterplots
#-----

runScatplt()
{
cat<<END>temp
                Parameters for SCATPLT
                *****

START OF PARAMETERS:
ppmt/ppmt${4}.out           -file with data
${1}  ${2}  14  0           -columns for X, Y, wt, third var.
-1.0e21  1.0e21            -trimming limits
${3}.ps                     -file for Postscript output
-5  5  0                   -X min and max, (0=arith, 1=log)
-5  5  0                   -Y min and max, (0=arith, 1=log)
1                            -plot every nth data point
0.4                         -bullet size: 0.1(sml)-1(reg)-10(big)
0.0  2.0                   -limits for third variable gray scale
PPMT: ${5} vs PPMT: ${6}   -title
END

echo temp | scatplt; rm temp
}

runScatplt 16 17 1 1  Ni SiO
runScatplt 16 18 2 1  Ni MgO
runScatplt 16 19 3 1  Ni Fe
runScatplt 16 20 4 1  Ni CaO
runScatplt 17 18 5 1  SiO MgO
runScatplt 17 19 6 1  SiO Fe
runScatplt 17 20 7 1  SiO CaO
runScatplt 18 19 8 1  MgO Fe
runScatplt 18 20 9 1  MgO CaO
runScatplt 19 20 10 1 Fe CaO

```

```

#-----
# Assemble a Single Plot of these Scatterplots
#-----
cat<<END>temp
                Parameters for PLOTTEM
                *****

START OF PARAMETERS:
ppmt/PPMT_scatter_plots.ps                -output file
2 5                                        -number of plots in X and Y
1.ps
2.ps
3.ps
4.ps
5.ps
6.ps
7.ps
8.ps
9.ps
10.ps
END
plotem temp; rm temp ; rm ?.ps ;rm 1?.ps

#-----
# Variograms of PPMT Factor-Gaussian Values
#-----
echo Performing variography of the PPMT-ncores for each factor...
# Test doing variography on the gaussian values and PPMT Factors.
# Choose from one of these files as Input to the variography:
# ppmt file: ppmt1.out
# ppmt normal scores: ppmt_nscore1.out
# Check Column numbers for variables in the ppmt output files!
# Here we calculate the column numbers for the NS values created by the PPMT process
v1=$((nvar+nvar+6))
v2=$((nvar+nvar+7))
v3=$((nvar+nvar+8))
v4=$((nvar+nvar+9))
v5=$((nvar+nvar+10))

echo Variable Column Numbers for PPMT Normal Scores:
echo Var1 = $v1
echo Var2 = $v2
echo Var3 = $v3
echo Var4 = $v4
echo Var5 = $v5
echo ' '

#maximum distances for variograms to be plotted dir1, dir2, dir3
maxdis1=200
maxdis2=200
maxdis3=20

#maximum y value for variogram plots
maxvariog=1.3

#join experimetal variogram points with dashed line 1=yes, 0=no
dash=0

#plot the variogram sill of 1 (NS Gaussian variable)? 1=yes, 2=no
sill=1

```

```

#-----
# Calculate Variograms
#-----
cat<<END>temp
                Parameters for VARCALC
                *****

START OF PARAMETERS:
ppmt/ppmt1.out          -file with data          ...
$xcol $ycol $zcol      -columns for X, Y, Z coordinates ...
$nvar $v1 $v2 $v3 $v4 $v5 -number of variables, column numbers (position...
-8.0 8.0               -trimming limits        ...
3                     -number of directions   ...
$az1 22.5 10 $dip1 22.5 10 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol ...
 30 25.0 12.5         -number of lags,lag distance,lag tolerance ...
$az2 22.5 10 $dip2 22.5 10 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol ...
 30 25.0 12.5         -number of lags,lag distance,lag tolerance ...
$az3 22.5 100 $dip3 22.5 100 0.0 -Dir 01: azm,azmtol,bandhorz,dip,diptol ...
 60 1.0 1             -number of lags,lag distance,lag tolerance ...
ppmt/PPMTNSgamv.out    -file for variogram output ...
0                     -legacy output (0=no, 1=write out gamv2004 ...
1                     -run checks for common errors ...
0                     -standardize sills? (0=no, 1=yes) ...
$nvar                 -number of variogram types ...
1 1 1 ?              -tail variable, head variable, variogram type...
2 2 1 ?              -tail variable, head variable, variogram type...
3 3 1 ?              -tail variable, head variable, variogram type...
4 4 1 ?              -tail variable, head variable, variogram type...
5 5 1 ?              -tail variable, head variable, variogram type...
END
cp temp work/audit_varcalc_ppmtF.par
echo temp | varcalc
echo Search Orientation 1 ${search[0]}
echo Search Orientation 2 ${search[1]}
echo Search Orientation 3 ${search[2]}

```

```

#-----
# Fit Variogram Factor/NS_1
#-----
cat<<END>temp

                Parameters for VARMODEL
                *****

START OF PARAMETERS:
ppmt/varmodell1.out      -file for modeled variogram points output ...
3                        -number of directions to model points along...
$az1  $dip1  1000  1     -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1     -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1   -azm, dip, npoints, point separation ...
2  0.05:0.06           -nst, nugget effect ...
1  0.75  $az1  $dip1  0   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    15   15   4         -a_hmax, a_hmin, a_vert (ranges) ...
1  0.20  $az1  $dip1  0   -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    500  600  5         -a_hmax, a_hmin, a_vert (ranges) ...
1  100000              -fit model (0=no, 1=yes), maximum iteration...
1.0                    -variogram sill (can be fit, but not ...
1                        -number of experimental files to use ...
ppmt/PPMTNSgamv.out    -experimental output file 1 ...
3 1 $((1+$nvar)) $((1+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10                 -# pairs weighting, inverse distance ...
0 10.0                 -fix Hmax/Vert anis. (0=no, 1=yes) ...
0 1.0                  -fix Hmin/Hmax anis. (0=no, 1=yes) ...
ppmt/varfitPPMT1.var   -file to save fit variogram model ...
END

echo temp | varmodel
rm temp

```

```

#-----
# Plot Factor/NS_1 Dir 1
#-----

cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v1.ps          -file for PostScript output      ...
2                  -number of variograms to plot   ...
0.0 $maxdis1       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 1 Dir 1 azm:$az1 dip:$dip1    -Title for variogram              ...
ppmt/PPMTNSgamv.out -1 file with variogram data        ...
1 $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodell1.out -2 file with variogram data        ...
1 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Factor/NS_1 Dir 2
#-----

cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v2.ps          -file for PostScript output      ...
2                  -number of variograms to plot   ...
0.0 $maxdis2       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 1 Dir 2 azm:$az2 dip:$dip2    -Title for variogram              ...
ppmt/PPMTNSgamv.out -1 file with variogram data        ...
$((1+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodell1.out -2 file with variogram data        ...
2 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Factor/NS_1 Dir 3
#-----

cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
ppmt/v3.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0  $maxdis3             -distance limits (from data if max<min) ...
0.0  $maxvariog           -variogram limits (from data if max<min) ...
$sill  1.0                -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 1 Dir 3 azm:$az3 dip:$dip3 -Title for variogram              ...
ppmt/PPMTNSgamv.out       -1 file with variogram data       ...
$(1+$nvar+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodell.out        -2 file with variogram data       ...
3  0  0  1  1            -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----

cat<<END>temp

    Parameters for PLOTTEM
    *****

START OF PARAMETERS:
ppmt/PPMT_variogram1.ps   -output file
1  3                      -number of plots in X and Y
ppmt/v1.ps                -first plot file
ppmt/v2.ps                -second plot file
ppmt/v3.ps                -third plot file
END

echo temp | plotem; rm temp
rm ppmt/v1.ps ppmt/v2.ps ppmt/v3.ps

```



```

#-----
# Fit Variogram Factor/NS_2
#-----

cat<<END>temp

                Parameters for VARMODEL
                *****

START OF PARAMETERS:
ppmt/varmodel2.out      -file for modeled variogram points output ...
3                      -number of directions to model points along...
$az1  $dip1  1000  1    -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1    -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1  -azm, dip, npoints, point separation ...
2      0.05:0.06      -nst, nugget effect ...
1      0.85  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
        10      10      3.4  -a_hmax, a_hmin, a_vert (ranges) ...
1      0.1  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
        350     300     3.5  -a_hmax, a_hmin, a_vert (ranges) ...
1      100000        -fit model (0=no, 1=yes), maximum iteration...
1.0    -variogram sill (can be fit, but not ...
1      -number of experimental files to use ...
ppmt/PPMTNSgamv.out   -experimental output file 1 ...
3 2  $((2+$nvar))  $((2+$nvar+$nvar))  -# of variograms (<=0 for all), variogram ...
1 1  10            -# pairs weighting, inverse distance ...
0      10.0        -fix Hmax/Vert anis. (0=no, 1=yes) ...
0      1.0         -fix Hmin/Hmax anis. (0=no, 1=yes) ...
ppmt/varfitPPMT2.var  -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Factor/NS_2 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v1.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis1              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 2 Dir 1 azm:$az1 dip:$dip1 -Title for variogram            ...
ppmt/PPMTNSgamv.out       -1 file with variogram data      ...
2 $dash 2 $dash 10        -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel2.out        -2 file with variogram data      ...
1 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Factor/NS_2 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v2.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis2              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 2 Dir 2 azm:$az2 dip:$dip2 -Title for variogram            ...
ppmt/PPMTNSgamv.out       -1 file with variogram data      ...
$(2+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel2.out        -2 file with variogram data      ...
2 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Factor/NS_2 Dir 3
#-----
cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
ppmt/v3.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0  $maxdis3             -distance limits (from data if max<min) ...
0.0  $maxvariog           -variogram limits (from data if max<min) ...
$sill      1.0            -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 2 Dir 3 azm:$az3 dip:$dip3 -Title for variogram              ...
ppmt/PPMTNSgamv.out       -1 file with variogram data       ...
$((2+$nvar+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel2.out       -2 file with variogram data       ...
3  0  0  1  1            -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
    1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
    7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
    13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
    19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

    Parameters for PLOTTEM
    *****

START OF PARAMETERS:
ppmt/PPMT_variogram2.ps   -output file
1  3                       -number of plots in X and Y
ppmt/v1.ps                -first plot file
ppmt/v2.ps                -second plot file
ppmt/v3.ps                -third plot file
END

echo temp | plotem; rm temp
rm ppmt/v1.ps ppmt/v2.ps ppmt/v3.ps

```

```

#-----
# Fit Variogram Factor/NS_3
#-----
cat<<END>temp

                Parameters for VARMODEL
                *****

START OF PARAMETERS:
ppmt/varmodel3.out      -file for modeled variogram points output ...
3                      -number of directions to model points along...
$az1  $dip1  1000  1    -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1    -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1  -azm, dip, npoints, point separation ...
2  0.1:0.11          -nst, nugget effect ...
1  0.55  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    16   15   5        -a_hmax, a_hmin, a_vert (ranges) ...
1  0.35  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    300  500  6        -a_hmax, a_hmin, a_vert (ranges) ...
1  100000          -fit model (0=no, 1=yes), maximum iteration...
1.0                -variogram sill (can be fit, but not ...
1                  -number of experimental files to use ...
ppmt/PPMTNSgamv.out -experimental output file 1 ...
3 3 $((3+$nvar)) $((3+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10             -# pairs weighting, inverse distance ...
0 10.0            -fix Hmax/Vert anis. (0=no, 1=yes) ...
0 1.0             -fix Hmin/Hmax anis. (0=no, 1=yes) ...
ppmt/varfitPPMT3.var -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Factor/NS_3 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v1.ps          -file for PostScript output          ...
2                  -number of variograms to plot        ...
0.0 $maxdis1       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 3 Dir 1 azm:$az1 dip:$dip1       -Title for variogram                  ...
ppmt/PPMTNSgamv.out -1 file with variogram data           ...
3 $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel3.out -2 file with variogram data           ...
1 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Factor/NS_3 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v2.ps          -file for PostScript output          ...
2                  -number of variograms to plot        ...
0.0 $maxdis2       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 3 Dir 2 azm:$az2 dip:$dip2       -Title for variogram                  ...
ppmt/PPMTNSgamv.out -1 file with variogram data           ...
$((3+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel3.out -2 file with variogram data           ...
2 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Factor/NS_3 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v3.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill 1.0                 -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 3 Dir 3 azm:$az3 dip:$dip3 -Title for variogram              ...
ppmt/PPMTNSgamv.out       -1 file with variogram data       ...
$((3+$nvar+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel3.out        -2 file with variogram data       ...
3 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
ppmt/PPMT_variogram3.ps   -output file
1 3                       -number of plots in X and Y
ppmt/v1.ps                -first plot file
ppmt/v2.ps                -second plot file
ppmt/v3.ps                -third plot file
END

echo temp | plotem; rm temp
rm ppmt/v1.ps ppmt/v2.ps ppmt/v3.ps

```

```

#-----
# Fit Variogram Factor/NS_4
#-----
cat<<END>temp

                Parameters for VARMODEL
                *****

START OF PARAMETERS:
ppmt/varmodel4.out      -file for modeled variogram points output ...
3                      -number of directions to model points along...
$az1  $dip1  1000  1    -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1    -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1  -azm, dip, npoints, point separation ...
2  0.05:0.06          -nst, nugget effect ...
1  0.75  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    12   15   4        -a_hmax, a_hmin, a_vert (ranges) ...
1  0.20  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    600  550  4.5      -a_hmax, a_hmin, a_vert (ranges) ...
1  100000             -fit model (0=no, 1=yes), maximum iteration...
1.0                  -variogram sill (can be fit, but not ...
1                    -number of experimental files to use ...
ppmt/PPMTNSgamv.out  -experimental output file 1 ...
3 4 $((4+$nvar)) $((4+$nvar+$nvar)) -# of variograms (<=0 for all), variogram ...
1 1 10               -# pairs weighting, inverse distance ...
0 10.0              -fix Hmax/Vert anis. (0=no, 1=yes) ...
0 1.0               -fix Hmin/Hmax anis. (0=no, 1=yes) ...
ppmt/varfitPPMT4.var -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Factor/NS_4 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v1.ps          -file for PostScript output          ...
2                  -number of variograms to plot        ...
0.0 $maxdis1       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 4 Dir 1 azm:$az1 dip:$dip1       -Title for variogram                  ...
ppmt/PPMTNSgamv.out -1 file with variogram data           ...
4 $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel4.out -2 file with variogram data           ...
1 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Factor/NS_4 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v2.ps          -file for PostScript output          ...
2                  -number of variograms to plot        ...
0.0 $maxdis2       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 4 Dir 2 azm:$az2 dip:$dip2       -Title for variogram                  ...
ppmt/PPMTNSgamv.out -1 file with variogram data           ...
$(4+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel4.out -2 file with variogram data           ...
2 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```



```

#-----
# Plot Factor/NS_4 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v3.ps                -file for PostScript output      ...
2                          -number of variograms to plot    ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill      1.0            -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 4 Dir 3 azm:$az3 dip:$dip3 -Title for variogram            ...
ppmt/PPMTNSgamv.out       -1 file with variogram data      ...
$((4+$nvar+$nvar)) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel4.out        -2 file with variogram data      ...
3 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
ppmt/PPMT_variogram4.ps   -output file
1 3                       -number of plots in X and Y
ppmt/v1.ps                -first plot file
ppmt/v2.ps                -second plot file
ppmt/v3.ps                -third plot file
END

echo temp | plotem; rm temp
rm ppmt/v1.ps ppmt/v2.ps ppmt/v3.ps

```

```

#-----
# Fit Variogram Factor/NS_5
#-----
cat<<END>temp

                Parameters for VARMODEL
                *****

START OF PARAMETERS:
ppmt/varmodel5.out      -file for modeled variogram points output ...
3                      -number of directions to model points along...
$az1  $dip1  1000  1    -azm, dip, npoints, point separation ...
$az2  $dip2  1000  1    -azm, dip, npoints, point separation ...
$az3  $dip3  500   0.1  -azm, dip, npoints, point separation ...
2    0.05:0.06         -nst, nugget effect ...
1    0.60  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    35    25    8        -a_hmax, a_hmin, a_vert (ranges) ...
1    0.35  $az1  $dip1  0  -it,cc,azm,dip,tilt (ang1,ang2,ang3) ...
    600   550   12       -a_hmax, a_hmin, a_vert (ranges) ...
1    100000           -fit model (0=no, 1=yes), maximum iteration...
1.0                      -variogram sill (can be fit, but not ...
1                          -number of experimental files to use ...
ppmt/PPMTNSgamv.out     -experimental output file 1 ...
3 5 $(5+$nvar) $(5+$nvar+$nvar) -# of variograms (<=0 for all), variogram ...
1 1 10                  -# pairs weighting, inverse distance ...
0 10.0                  -fix Hmax/Vert anis. (0=no, 1=yes) ...
0 1.0                   -fix Hmin/Hmax anis. (0=no, 1=yes) ...
ppmt/varfitPPMT5.var   -file to save fit variogram model ...
END

echo temp | varmodel

```

```

#-----
# Plot Factor/NS_5 Dir 1
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v1.ps          -file for PostScript output          ...
2                  -number of variograms to plot          ...
0.0 $maxdis1       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 5 Dir 1 azm:$az1 dip:$dip1       -Title for variogram                  ...
ppmt/PPMTNSgamv.out -1 file with variogram data            ...
5 $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel5.out -2 file with variogram data            ...
1 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Plot Factor/NS_5 Dir 2
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v2.ps          -file for PostScript output          ...
2                  -number of variograms to plot          ...
0.0 $maxdis2       -distance limits (from data if max<min) ...
0.0 $maxvariog     -variogram limits (from data if max<min) ...
$sill 1.0          -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 5 Dir 2 azm:$az2 dip:$dip2       -Title for variogram                  ...
ppmt/PPMTNSgamv.out -1 file with variogram data            ...
$(5+$nvar) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel5.out -2 file with variogram data            ...
2 0 0 1 1         -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

```

```

#-----
# Plot Factor/NS_5 Dir 3
#-----
cat<<END>temp

Parameters for VARPLOT (modified version of vargplt2008)
*****

START OF PARAMETERS:
ppmt/v3.ps                -file for PostScript output          ...
2                          -number of variograms to plot        ...
0.0 $maxdis3              -distance limits (from data if max<min) ...
0.0 $maxvariog            -variogram limits (from data if max<min) ...
$sill      1.0            -plot sill (0=no,1=yes,2+=multiple), sill...
PPMT 5 Dir 3 azm:$az3 dip:$dip3      -Title for variogram                  ...
ppmt/PPMTNSgamv.out        -1 file with variogram data           ...
$( (5+$nvar+$nvar) ) $dash 2 $dash 10 -variogram #, dash #, pts?, line?, color ...
ppmt/varmodel5.out        -2 file with variogram data           ...
3 0 0 1 1                 -variogram #, dash #, pts?, line?, color ...

Color Codes for Variogram Lines/Points:
1=red, 2=orange, 3=yellow, 4=light green, 5=green, 6=light blue,
7=dark blue, 8=violet, 9=white, 10=black, 11=purple, 12=brown,
13=pink, 14=intermediate green, 15=gray 16=gray10, 17=gray20, 18=gray30,
19=gray40, 20=gray50, 21=gray60 22=gray70, 23=gray80, 24=gray90

END

echo temp | varplot
rm temp

#-----
# Assemble a Plotfile for these variograms
#-----
cat<<END>temp

Parameters for PLOTTEM
*****

START OF PARAMETERS:
ppmt/PPMT_variogram5.ps   -output file
1 3                       -number of plots in X and Y
ppmt/v1.ps                -first plot file
ppmt/v2.ps                -second plot file
ppmt/v3.ps                -third plot file
END

echo temp | plotem; rm temp
rm ppmt/v1.ps ppmt/v2.ps ppmt/v3.ps

#clean up
rm VFcol VFcol1 VFmin VFmax VFRank NiCol temp*
rm ImpData1.dat VFname
date
echo "Sim3_Impute.bsh finished"
#=====
# Script complete
#=====

#*****

```

```

#!/bin/bash
#=====
# Script Sim4.1_Simulate.bsh
#=====
# Simulation
#=====
#read Master_Parameter List

echo "Sim4_Simulate.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt

grid_definition_file=$grid_definition_file
nreal=$nrealisations
nvar=$nvar
keepcode=$keep_code
keyoutfile=$rock
xcol=$xcol
ycol=$ycol
zcol=$zcol

#conditioning data
infile=ppmt/ppmt1.out
#number of data to include in simulation
ndata_sim=$ndata_sim
#remember to change the random seed...
random=$random_seed

date

#-----
# Run calculator to add the ROCK column to infile
#-----
cat<<END>temp
        Parameters for CALCULATOR
        *****

START OF PARAMETERS:
ppmt/ppmt1.out          -file with data
ppmt/ppmt2.out          -output file
-1.0e21  1.0e21         -trimming limits
1                        -output columns
Rock                    -new variable names
1                        -number of functions
(v1*0.0/100)+$keepcode
END

echo temp | calculator
rm temp

```

```

#-----
# Conditional Simulation of PPMT Factors using USGSIM
#-----

#we need variogram sill values - calculate and save them here
#awk select lines with it OR nugget and sum the second field value
awk '/it|nugget/ {sum += $2} END{print sum}' ppmt1/varfitPPMT1a.var > work/sill1
awk '/it|nugget/ {sum += $2} END{print sum}' ppmt1/varfitPPMT2a.var > work/sill2
awk '/it|nugget/ {sum += $2} END{print sum}' ppmt1/varfitPPMT3a.var > work/sill3
awk '/it|nugget/ {sum += $2} END{print sum}' ppmt1/varfitPPMT4a.var > work/sill4
awk '/it|nugget/ {sum += $2} END{print sum}' ppmt1/varfitPPMT5a.var > work/sill5

#add variogram sill value into variogram description - lines with 'nugget'
echo $keepcode 1 1 >temp

awk -v sill=$(for ((x=1 ; x<=$nvar ; x++)); do
let v${x}=$last-$nvar+${x}-1
done

```

```

cat<<END>temp0
      Parameters for USGSIM
      *****

START OF MAIN:
$nreal          -number of realizations to generate, 0=kriging
$nvar           -number of variables being simulated
1              -number of rock types to consider
$random        -random number seed
END

cat rock/$grid_definition_file>>temp0
cat<<END>templ
sims/sgsim.gsb  -file for simulation output
2              -output format: (0=reg, 1=coord, 2=binary)
sims/Imputed_sgsim.out -file for imputed values in case of heterotopic...
3              -debugging level: 0,1,2,3
sims/sgsim.dbg  -file for debugging output
...
...

START OF SRCH:
$nndata_sim     -number of data to use per variable
${radii[0]} ${radii[1]} ${radii[2]} -maximum search radii (hmax,hmin,vert)
${search[0]} ${search[1]} ${search[2]} -angles for search ellipsoid
0              -sort by distance (0) or covariance (1)
$keepcode 1 1  -if sorting by covariance, indicate variogram
...
...

START OF DATA:
ppmt/ppmt2.out -file with primary data
$xcol $ycol $zcol 4 $last -columns for X,Y,Z,wt,rock type
$vv1 $vv2 $vv3 $vv4 $vv5 -columns for variables
0              -clip data to grid, 1=yes
1              -assign to the grid, 0=none, 1=nearest, 2=avera...
-99.0          99.0 -trimming limits
...
...

START OF TRND:
0              -consider ordinary kriging (0=no, 1=yes)
0              -number of locally varying means
1 1           -rock type number, primary variable number
lvml101.dat    -file with local mean
1             -column for value
1 2           -rock type number, primary variable number
lvml102.dat    -file with local mean
1             -column for value
...
...

START OF ROCK:
$keepcode     -rock type codes
rock/$keyoutfile -file with rock types
1             -column for rock type
...
...

START OF MULT:
1             -1=independent, 2=collocated, 4=full cokriging,...
0             -option for primary variables if BU / for secon...
1             -perform cosimulation for multiple primary vari...
$keepcode     -rock type for correlation matrix
 1 0.0 0.0 0.0 0.0 -correlation matrix: 1-1, 1-2, 1-3, 1-sec1, 1-s...
0.0 1 0.0 0.0 0.0 -2-1, 2-2, 2-3, 2-sec1, 2-sec2
0.0 0.0 1 0.0 0.0 -...
0.0 0.0 0.0 1 0.0 -...
0.0 0.0 0.0 0.0 1 -...
...
...

START OF VARG:
$nvar         -number of variograms
END

```

```
cat temp1>>temp0
cat tempv>>temp0; cp temp0 sims/audit_usgsim.par
echo temp0 | usgsim336
rm temp0; rm temp1; rm tempv
date
echo "Sim4.1_Simulate.bsh finished"
#=====
# Script complete
#=====
#*****
```



```

#!/bin/bash
#=====
# Script Sim5_Backtransform.bsh
#=====
# PPMT back-transform
#=====
#read Master_Parameter List
# we need to record the number of simulated points and divide this into nx,ny and nz
# we need to be able to divide it up into three factors. Also include nreals
# Read Master Parameter
List=====
echo "Sim5_Backtransform.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt
grid_definition_file=$grid_definition_file
nreal=$nrealisations
date
echo reading simulation grid definition...
awk 'NR==1{print $1}' rock/$grid_definition_file > work/nx
awk 'NR==2{print $1}' rock/$grid_definition_file > work/ny
awk 'NR==3{print $1}' rock/$grid_definition_file > work/nz
#-----
# Reverse the PPMT transform
#-----
cat<<END>temp
                PPMT Back Transformation
                *****

START OF PARAMETERS:
ppmt/ppmt1.trn      -file with transformation data
sims/sgsim.gsb     -file with data to back-transform (GSB ...)
$nvar 1 2 3 4 5    -nvar, column numbers
-99 99            -trimming limits
$(<work/nx) $(<work/ny) $(<work/nz) $nreal -nx, ny, nz, nreal (ignored if nx = 0)
ppmt/ppmt_b.gsb   -file for back transformed values
0                 -enforce N(0,1)? (0=no,1=yes)
0                 -consider local uncertainty? (0=no,1=yes)
kt3dv.out         -local file
1 2               -columns for kriging variances
.2                -weight factor
1                 -number of loops
END
echo temp | ppmt_b
rm temp
date
echo "Sim5_Backtransform.bsh finished"
#=====
# Script complete
#=====

#*****

```

```

#!/bin/bash
#=====
# Script Sim6_Sim_Histograms.bsh
#=====
# Build histograms of simulation
#=====

#read Master_Parameter List
echo "Sim6_Sim_Histograms.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt

#Simulation Results....
nreal=$nrealisations
simfile=ppmt/ppmt_b.gsb
sample=-1          # use all the data
qdisplay=1000      # limit no. of display points
gsb_rockmod=$rock_nreal #use multiple realisation version here...
lithcode=$keep_code
grid_definition=$grid_definition_file

#these should not change
infile=work/declus_nn.out
nvar=5
var1=7
var2=8
var3=9
var4=10
var5=11
wtcol=12
lithcol=5

#read the grid sizes

awk 'NR==1{print $1}' rock/$grid_definition > work/nx
awk 'NR==2{print $1}' rock/$grid_definition > work/ny
awk 'NR==3{print $1}' rock/$grid_definition > work/nz

```

```

#-----
# Prepare variable list with Casebuild
#-----
cat<<END>temp
        Parameters for CASEBUILD
        *****

START OF PARAMETERS:
$infile                -file with data
$nvar  $var1 $var2 $var3 $var4 $var5  -number of variables and columns
-998 100.              -trimming limits
work/settings.sh      -file for output case statement
END
echo temp |casebuild
echo
dos2unix work/settings.sh
echo Locating Min and Max Values
# put mins and max values into variable files
for ((x=1 ;x<=$nvar ;x++));do
. work/settings.sh ${x}
echo ${min} > work/tempmin${x}
echo ${max} >> work/tempmin${x}
echo ${name} > work/names${x}
cat work/tempmin${x} | tr '\n' ' ' > work/limits${x};
echo Minimum and Maximum Values for ${name} = ${(<work/limits${x})}
done
rm work/tempmin*

```

```

#-----
# 1st Variable:col=1 in simulation file
#-----

echo processing histograms for $(

```

```

#-----
# 2nd Variable:col=2 in simulation file
#-----

echo processing histograms for $(

```

```

#-----
# 3rd Variable:col=3 in simulation file
#-----

echo processing histograms for $(

```

```

#-----
# 4th Variable:col=4 in simulation file
#-----

echo processing histograms for $(

```

```

#-----
# 5th Variable:col=5 in simulation file
#-----

echo processing histograms for $(

```



```

#!/bin/bash
#=====
# Script Sim7A_Sims_Variograms.bsh
#=====
# Calculate variograms of simulation
#=====

#read Master_Parameter List
echo "Sim7A_Sims_Variograms.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt

#Simulation Results....
grid_definition_file=$grid_definition_file
nreal=$nrealisations
nvar=$nvar
simfile=ppmt/ppmt_b.gsb
sample=-1
qdisplay=10000
keepcode=$keep_code
gsb_keyout_file=$rock_nreal

date

```

```

#-----
# Calculate variograms from realisations. Must adjust idx(1-3),idy(1-3) and idz(1-3)
#-----
cat<<END>>temp
      Parameters for VARSIM
      *****

START OF PARAMETERS:
rock/$gsb_keyout_file           -file with lithology information
1 $keepcode                     -lithology column (0=not used), code
$simfile                        -file with data
$nvar 1 2 3 4 5                 -number of variables, column numbers
-1 100.0                        -trimming limits
sims/varsim_reals_gsb.out       -output file for variograms of realizations
sims/varsim_avg_gsb.out        -output file for average variogram
END

cat rock/$grid_definition_file>>temp

cat<<END>>temp
$nreal                          -nrealisations
3 160                           -number of directions, number of lags
${ixd_1} ${iyd_1} ${izd_1}      -ixd(1),iyd(1),izd(1)
${ixd_2} ${iyd_2} ${izd_2}      -ixd(2),iyd(2),izd(2)
${ixd_3} ${iyd_3} ${izd_3}      -ixd(3),iyd(3),izd(3)
0                                -standardize sill? (0=no, 1=yes)
5                                -number of variograms
1 1 1                           -tail variable, head variable, variogram type
2 2 1                           -tail variable, head variable, variogram type
3 3 1                           -tail variable, head variable, variogram type
4 4 1                           -tail variable, head variable, variogram type
5 5 1                           -tail variable, head variable, variogram type

type 1 = traditional semivariogram
     2 = traditional cross semivariogram
     3 = covariance
     4 = correlogram
     5 = general relative semivariogram
     6 = pairwise relative semivariogram
     7 = semivariogram of logarithms
     8 = semimadogram
     9 = indicator semivariogram - continuous
    10= indicator semivariogram - categorical

VarSim is a conversion of gamsim_ave to support
the file format used by varcalc, varmodel and varplot.
END

echo temp | varsim; rm temp

date

echo "Sim7A_Sims_Variograms.bsh finished"
#-----
# Script complete
#-----

#*****

```

```

#!/bin/bash
#=====
# Script Sim7C_Plot_Variograms_vertical5m.bsh
#=====
# Plot variograms
#=====
#read Master_Parameter List
echo "Sim7C_Plot_Variograms.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    chmod 777 PPMT_Master_Parameters.txt
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt
drill_data=$input_data
nvar=$nvar
var1=$var1
var2=$var2
var3=$var3
var4=$var4
var5=$var5
nreal=$nrealisations
#orientation of principal axes
#principal directions for variogram calculation-read from files -load wih error checks
if [ -e work/azm1 ] ; then
az1=$(cat work/azm1)
echo "reading variogram orientation data"
else
echo "Fatal Error: azm1 file not found"
exit 1
fi
if [ -e work/dip1 ] ; then
dip1=$(cat work/dip1)
else
echo "Fatal Error: dip1 file not found"
fi
if [ -e work/azm2 ] ; then
az2=$(cat work/azm2)
else
echo "Fatal Error: azm2 file not found"
exit 1
fi
if [ -e work/dip2 ] ; then
dip2=$(cat work/dip2)
else
echo "Fatal Error: dip2 file not found"
exit 1
fi
if [ -e work/azm3 ] ; then
az3=$(cat work/azm3)
else
echo "Fatal Error: azm3 file not found"
exit 1

```

```

fi
if [ -e work/dip3 ]; then
dip3=$(cat work/dip3)
else
echo "Fatal Error: dip3 file not found"
exit 1
fi

#-----
# Calculate declustered variance of data from nearest neighbour decuster results
#-----

dchistout=eda/NN_Declus.ps #replace the previous one
filex=work/declus_nn.out
#what are the trimming limits on these variables?
trimlow=-0.00000001
trimhigh=100
wtcol=$(( $var5+1))

#-----
# Run histograms
#-----
runHistplt()
{
cat<<END >temp
                Parameters for HISTPLT
                *****

START OF PARAMETERS:
$filex          -file with reference data
${col} $wtcol   -columns for variable and weight (set ivr<0 to ...
${trimlow} ${trimhigh} -trimming limits ...
${x}.ps        -file for PostScript output ...
0.0 -1        -attribute minimum and maximum (<0 for automati...
-1.0          -frequency maximum (<0 for automatic) ...
50            -number of classes (<0 to label bars with numbe...
0            -0=arithmetic, 1=log scaling ...
0            -0=frequency, 1=cumulative histogram, 2=consta...
20           -number of cum. quantiles (<0 for all) or the c...
-2           -number of decimal places (<0 for auto.) ...
${header}    -title ...
default     -X label, if set to default the variable in...
Frequency   -Y label, if set to default the variable in...
1e21        -reference value for box plot ...
1.5         -horizontal positioning of stats (L to R: -1 to...
2           -reference stats to plot(0=none,1=minimal,2=ful...
reals.out   -file for realization data (if plotting realiza...
1 0        -columns for variable and weight ...
10         -nreal ...
10 100 1   -nx ny nz ...
END

echo temp | histplt2008
}
N=$nvar
for ((x=1 ; x<=N ; x++)); do
    . work/settings1.sh ${x}
    header="${name} Declustered PDF"
    runHistplt
done

```

```

#-----
# Group histograms plots
#-----

cat<<END>temp

                Parameters for PLOTEM
                *****

START OF PARAMETERS:
$dchistout
2 3
END
for ((x=1 ; x<=N ; x++)); do

    echo "${x}.ps" >>temp
done
    plotem temp
    rm temp ?.ps ??.ps junk.out

#-----
# Back-transform the NS variograms of the input data
#-----
#use this step to validate the variogram of the simulations against the BT NS vgrams
runVgramp()
{
cat<<END> temp

                Parameters for VMODELYZ
                *****

START OF PARAMETERS:
vgram/${name1}BT.var                -file for variogram output
3 200                                -number of directions and lags
${az1} ${dip1} 1.0                  -azm, dip, lag distance
${az2} ${dip2} 1.0                  -azm, dip, lag distance
${az3} ${dip3} 0.2                  -azm, dip, lag distance
nscore/unscore${x}.trn              -file with input transformation table
0.0 ${max1}                          -minimum and maximum data value
1 ${min1}                             -lower tail option and parameter
1 ${max1}                             -upper tail option and parameter
${<work/dc_variance_${name1}}        -variance of data (watch declustering)*
END

cat vgram/varfit${name1}.var >> temp

echo temp | vmodelyz

cp temp vgram/vmodelyz_${name1}.par
}

#loop controlling vmodelyz
N=$nvar
for ((x=1 ; x<=N ; x++)); do
    . work/settings1.sh ${x}
    name1=${name}
    max1=${max}
    min1=${min}
    runVgramp
done

```

```

#-----
# Plot Variograms of Simulations overlain with BT NS Variograms - Direction 1
#-----

runSimvar()
{
echo ${start1} ${step} ${total}
cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
${output1}                -file for PostScript output
${((2+$nreal))}           -number of variograms to plot
0.0  200                  -distance limits (from data if max<min)
0.0  ${yaxis}             -variogram limits (from data if max<min)
1    $(<work/dc_variance_${name}) -plot sill (0=no,1=yes,2+=multiple), sill value(s))
${title1}                 -Title for variogram
END

for ((i=${start1};i<=${total};i+=${step}));
do
echo sims/varsim_reals_gsb.out >> temp
echo $i 0 0 1 24 >> temp
done

#Use this option to validate against the BT NS vgrams
echo vgram/${name}BT.var >> temp
echo 1 0 0 1 1 >> temp

#This is to plot the raw variogram declustered variance
echo vgram/varmodel${name}_declust_var.out >> temp
echo 1 0 0 0 7 >> temp

echo temp | varplot
cat temp > varplot.par

rm temp

echo ${start2} ${step} ${total}

```

```

#-----
# Plot Variograms of Simulations overlain with BT NS Variograms - Direction 2
#-----

cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
${output2}                -file for PostScript output
${((2+$nreal))}           -number of variograms to plot
0.0  200                  -distance limits (from data if max<min)
0.0  ${yaxis}             -variogram limits (from data if max<min)
1    $(<work/dc_variance_${name}) -plot sill (0=no,1=yes,2+=multiple), sill value(s)
${title2}                 -Title for variogram
END

for ((i=${start2};i<=${total};i+=${step}));
do
echo sims/varsim_reals_gsb.out >> temp
echo $i 0 0 1 24 >> temp
done

#Use this option to validate against the BT NS vgrams
echo vgram/${name}BT.var >> temp
echo 2 0 0 1 1 >> temp

#This is to plot the raw variogram declustered variance
echo vgram/varmodel${name}_declust_var.out >> temp
echo 2 0 0 0 7 >> temp

echo temp | varplot
rm temp

echo ${start3} ${step} ${total}

```

```

#-----
# Plot Variograms of Simulations overlain with BT NS Variograms - Direction 3
#-----

cat<<END>temp

    Parameters for VARPLOT (modified version of vargplt2008)
    *****

START OF PARAMETERS:
${output3}                -file for PostScript output
${((2+$nreal))}           -number of variograms to plot
0.0  5.0                  -distance limits (from data if max<min)
0.0  ${yaxis}             -variogram limits (from data if max<min)
1    ${<work/dc_variance_${name}} -plot sill (0=no,1=yes,2+=multiple), sill value(s)
${title3}                 -Title for variogram
END

for ((i=${start3};i<=${total};i+=${step}));
do
echo sims/varsim_reals_gsb.out >> temp
echo $i 0 0 1 24 >> temp
done

#Use this option to validate against the BT NS vgrams
echo vgram/${name}BT.var >> temp
echo 3 0 0 1 1 >> temp

#This is to plot the raw variogram declustered variance
echo vgram/varmodel${name}_declust_var.out >> temp
echo 3 0 0 0 7 >> temp

echo temp | varplot
rm temp

#-----
# Group Variograms into 1 Plot File
#-----

cat<<END>temp

    Parameters for PLOTEM
    *****

START OF PARAMETERS:
sims/${output0}           -output file
1  3                      -number of plots in X and Y
${output1}
${output2}
${output3}

END

echo temp | plotem; rm temp

rm "${name} 1_sim.ps"
rm "${name} 2_sim.ps"
rm "${name} 3_sim.ps"
}

```



```

for((x=1; x<=$nvar ; x++));do
  . work/settings1.sh ${x}
  output0="${name}_sim_variograms_vertical5m.ps"
  output1="${name}_1_sim.ps"
  output2="${name}_2_sim.ps"
  output3="${name}_3_sim.ps"
  title1="${name} Dir 1 azim:${az1} dip:${dip1}"
  title2="${name} Dir 2 azim:${az2} dip:${dip2} "
  title3="${name} Dir 3 azim:${az3} dip:${dip3}"
  step=$((($nvar*3))
  total=$((($step*$nreal))
  start1=${x}
  start2=$((start1+$nvar))
  start3=$((start2+$nvar))
  #dsill=1 #option to validate against raw inverse correlograms
  dsill=work/dc_variance_${name} #option to validate against BT NS/Raw variograms
  yaxis=$(echo "scale=3; 1.25*${sdev}*${sdev}" | bc)
  runSimvar
done
rm varplot.par

echo "Sim7C_Plot_Variograms.bsh finished"
#=====
# Script complete
#=====

#*****

```

```

#!/bin/bash
#=====
# Script Sim8.1_Ascii_output.bsh
#=====
# Output simulations into .csv files
#=====

#read Master_Parameter List
echo "Sim8_Ascii_output.bsh"
echo "Attempting read of Master Parameter File..."
if [ -e PPMT_Master_Parameters.txt ]
then
    echo "Read OK..."
    dos2unix PPMT_Master_Parameters.txt
else
    echo "+++++++ FATAL ERROR: Parameter File Not Found - ID10T error ++++++"
    exit 1
fi
# source the parameter file
. PPMT_Master_Parameters.txt

nreal=$nrealisations
nvar=$nvar
ppmt_output=ppmt/ppmt_b.gsb
keep_code=$keep_code
rockmodel=$rock
grid_definition=$grid_definition_file

```

```

#-----
# Run GSB2ASCII to create .csv files
#-----
#output each variable separately because of memory constraints
#uses gsb2ascii.exe [need version 1.3.1] to work with single realisation gsb file
dumpAscii()
{
echo " "
echo "Processing simulation output for variable  "${name} : ${x} of $nvar
echo " "
cat<<END>temp0
    Parameters for GSB2ASCII
    *****
START OF PARAMETERS:
$ppmt_output      -input GSB file with data to convert
1 ${x}            -number of variable and columns
0 -99             -variable trimming option code and default value (see note)
-0.01  101.0     -trimming limits
2                -block trimming option (see note)
1                -if 1, trimming variable number
rock/$rockmodel  -if 2, file with keyout array
1 $keep_code     -if 2, keyout column and value to keep
0                -if >1, append original grid index to output? (0=no,1=yes)
END
# truncation using single precision arithmetic no longer a problem!
awk 'NR==1' rock/$grid_definition| awk 'BEGIN{OFMT="%.3f"}{ print $1, $2, $3 }' >temp00
awk 'NR==2' rock/$grid_definition| awk 'BEGIN{OFMT="%.3f"}{ print $1, $2, $3 }' >>temp00
awk 'NR==3' rock/$grid_definition| awk 'BEGIN{OFMT="%.3f"}{ print $1, $2, $3 }' >>temp00
cat temp00 >> temp0
cat<<END>>temp0
$nreal           -number of realizations (0=number of reals in datafl)
${output}        -output ASCII file
2 2 1           -option codes for header, realizations and delimiter
(f18.5)         -format specifier in fortran notation
1               -remove leading blanks and trailing zeros/decimals? (0=no,1=yes)
END
echo temp0 | gsb2ascii
rm temp00
rm temp0
}
for((x=1; x<=$nvar ; x++));do
    . work/settings.sh ${x}
    output="sims/simres${name}.csv"
    dumpAscii
done
#if outputting realisations to individual files [2 2 1] must edit variable names
for file in sims/simres*.csv #loop over all csvs in the sims directory
do
echo 'processing: ' $file
echo ${file:11} | awk '{split($0, a,","); print a[1]}' > newname
echo $(<newname)
awk -F"," 'NR==1{print $4}' $file > oldname
echo $(<oldname)
sed -i "s/${<oldname}/${<newname}/g" $file
done
echo "Sim8.1_Ascii_output.bsh finished"
#=====
# Script complete
#=====

```

## APÊNDICE E – RSTUDIO SCRIPT

Os painéis de lavra foram gerados por meio de clusterização de médias k, conforme script abaixo.

```

---
title: "k-means cluster"
author: "Paulo Faria"
date: "May 2019"
output:
  beamer_presentation:
    colortheme: beaver
    theme: Boadilla
theme: Boadilla
colortheme: beaver
fontsize: 8pt
editor_options:
  chunk_output_type: console
---
\frametitle{Preamble}
Install libraries
```{r loading library, message=FALSE, warning=FALSE}
setwd("E:/R/packages")
install.packages("data.table_1.12.0.zip", repos = NULL, type = "win.binary")
install.packages("plotrix_3.7-5.zip", repos = NULL, type = "win.binary")
library(data.table)
library(lattice)
library(plotrix)
```

Define work directory and load the file for processing.
```{r}
setwd("E:/PPMT")
md5=fread("md5sim_reg.csv", sep=",", na.strings="-")
```

Run k-means and flag clusters to the file.
```{r echo=TRUE}
#Export as jpeg file (with this option there is no preview in the plot pane)
jpeg(file="kmeans70.jpg",height=2000, width=3000, units="px",quality = 100,pointsize = 12,type =
c("cairo"))
# define a matrix to view (this allows to group the plots)
par(mfcol=c(7,8))
#Loop over k-means
for (i in c(1:50)) {
RNGkind(kind = NULL,normal.kind = NULL)
#For four-week panel change 70 to 40 in the line below
clusters <- kmeans(md5[,2:3],70,iter.max = 1000,nstart = 50)
str(clusters)
md5$kmean <- as.factor(clusters$cluster)
print(plot(YC~XC,md5, main = paste0("kmean",i), col=md5$kmean))
setnames(md5,"kmean", paste0("kmean",i))
}
#End the process of writing jpeg file
dev.off()
```

Save csv file
```{r}
write.csv(md5, file = "md5kmean70.csv")
```

```

## APÊNDICE F – DATAMINE SCRIPT

Os cálculos dos coeficientes da variação e categorização final foram feitos por meio dos scripts abaixo no software Datamine.

```
//=====
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<HTML>
<HEAD>
<META Name="DatamineScript" Content="Version3" />
<TITLE>Command Automatically recorded html script.</TITLE>
<SCRIPT TYPE="text/javascript">

// -----

//Script criada por Paulo Henrique Faria - Revisada em Junho/2019

//Standard initialisation for Datamine Studio

var oDmApp,oScript,dmdir,oDmBrowser,oDmFile,fso;

oDmApp= null;
oScript = null;
oScript = new ActiveXObject("StudioCommon.ScriptHelper");
oScript.initialize(window);
oDmApp = oScript.getApplication();
dmdir=oDmApp.ActiveProject.Folder;
oDmBrowser = oDmApp.ActiveProject.Browser;
oDmFile = new ActiveXObject("DmFile.DmTableADO");
fso = new ActiveXObject("Scripting.FileSystemObject");
var ForReading=1, ForWriting=2, ForAppending=8;
var oDmApp = null;
var oScript = null;
var dmdir
function AutoConnect() {
    try {
        oScript = new ActiveXObject("StudioCommon.ScriptHelper");
        oScript.initialize(window);
        oDmApp = oScript.getApplication();
        //Attempt to Use the Active Datamine Session
        if (oDmApp == null || oDmApp.ActiveProject == null)
        {
            alert("There are no active Studio3 projects open.\n Please open a Studio 3
project before continuing.");
            window.close(); // Closes the script window
            return false;
        } else
            return true;
    }
    catch (e) {
        alert("Failed\nReason: " + e.description);
        if (oDmApp) oDmApp.Quit(); // release the session to close it down
    }
    return false;
}

var oDmFile = new ActiveXObject("DmFile.DmTableADO");
```

```

// =====
//                               Calcula CATEG com base em Coeficiente de Variação
// =====

function btnExecute_CategCoV() {

//Entradas
var md5kmean50 = "md5kmean70"; // arquivo com clusters
var mdreg = "md5sim_reg"; // arquivo de modelo
var nkmean = 50; // número de realizações de cluster kmeans
var analitos = ["Ni", "Fe", "SiO2", "CaOMgO"]; // analitos em estudo
var nsim = 50 // Número de realizações
var threshold = 0.05 // Limite considerado como aceitável para variação
var rate = 0.9 // Proporção de vezes em que o CV deve ser menor que o threshold

//Saídas
var mdout = "md5sim_categ_kmean70"; // arquivo de modelo com simulações

//=====

//Cria arquivo para looping final

oDmApp.ParseCommand("selcop &IN=" + md5kmean50 + " &OUT=temp_loop *F1=IJK @KEEPALL=1");

for (var k = 1; k <= nkmean; k++) {

//Acumula teores de cada realização individualmente em cada realização kmean

oDmApp.ParseCommand("selcop &IN=" + md5kmean50 + " &OUT=temp *F1=IJK *F2=kmean" + [k] + ""+
" @KEEPALL=1");

oDmApp.ParseCommand("join &IN1=" + mdreg + " &IN2=temp &OUT=temp1 *KEY1=IJK @SUBSETR=0"+
" @SUBSETF=0 @CARTJOIN=0 @PRINT=0");

oDmApp.ParseCommand("extra &IN=temp1 &OUT=temp2 @APPROX=0 @PRINT=0 'COUNT=1' 'GO'");

oDmApp.ParseCommand("mgsort &IN=temp2 &OUT=temp3 *KEY1=kmean" + [k] + " @ORDER=1");

oDmApp.ParseCommand("accmlt &IN=temp3 &OUT=temp4 *KEY1=kmean" + [k] + " @ALLRECS=0
@UNSORTED=0");

//Calcula média, desvio padrão e coeficiente de variação

for (var i = 0; i < analitos.length; i++) {

var j = 1;

oDmApp.ParseCommand("extra &IN=temp4 &OUT=temp5 @APPROX=0 @PRINT=0"+

//Divide valores acumulados pelo contador "COUNT"

" ' + analitos[i] + " + [j] + "=" + analitos[i] + " + [j] + "/COUNT'+
" ' "+ analitos[i] + " + [j+1] + "=" + analitos[i] + " + [j+1] + "/COUNT'+
" ' "+ analitos[i] + " + [j+2] + "=" + analitos[i] + " + [j+2] + "/COUNT'+
" ' "+ analitos[i] + " + [j+3] + "=" + analitos[i] + " + [j+3] + "/COUNT'+
" ' "+ analitos[i] + " + [j+4] + "=" + analitos[i] + " + [j+4] + "/COUNT'+
" ' "+ analitos[i] + " + [j+5] + "=" + analitos[i] + " + [j+5] + "/COUNT'+
" ' "+ analitos[i] + " + [j+6] + "=" + analitos[i] + " + [j+6] + "/COUNT'+
" ' "+ analitos[i] + " + [j+7] + "=" + analitos[i] + " + [j+7] + "/COUNT'+

```

```

" ' "+ analitos[i] + "" + [j+8] + "=" + analitos[i] + "" + [j+8] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+9] + "=" + analitos[i] + "" + [j+9] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+10] + "=" + analitos[i] + "" + [j+10] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+11] + "=" + analitos[i] + "" + [j+11] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+12] + "=" + analitos[i] + "" + [j+12] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+13] + "=" + analitos[i] + "" + [j+13] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+14] + "=" + analitos[i] + "" + [j+14] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+15] + "=" + analitos[i] + "" + [j+15] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+16] + "=" + analitos[i] + "" + [j+16] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+17] + "=" + analitos[i] + "" + [j+17] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+18] + "=" + analitos[i] + "" + [j+18] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+19] + "=" + analitos[i] + "" + [j+19] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+20] + "=" + analitos[i] + "" + [j+20] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+21] + "=" + analitos[i] + "" + [j+21] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+22] + "=" + analitos[i] + "" + [j+22] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+23] + "=" + analitos[i] + "" + [j+23] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+24] + "=" + analitos[i] + "" + [j+24] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+25] + "=" + analitos[i] + "" + [j+25] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+26] + "=" + analitos[i] + "" + [j+26] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+27] + "=" + analitos[i] + "" + [j+27] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+28] + "=" + analitos[i] + "" + [j+28] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+29] + "=" + analitos[i] + "" + [j+29] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+30] + "=" + analitos[i] + "" + [j+30] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+31] + "=" + analitos[i] + "" + [j+31] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+32] + "=" + analitos[i] + "" + [j+32] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+33] + "=" + analitos[i] + "" + [j+33] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+34] + "=" + analitos[i] + "" + [j+34] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+35] + "=" + analitos[i] + "" + [j+35] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+36] + "=" + analitos[i] + "" + [j+36] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+37] + "=" + analitos[i] + "" + [j+37] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+38] + "=" + analitos[i] + "" + [j+38] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+39] + "=" + analitos[i] + "" + [j+39] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+40] + "=" + analitos[i] + "" + [j+40] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+41] + "=" + analitos[i] + "" + [j+41] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+42] + "=" + analitos[i] + "" + [j+42] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+43] + "=" + analitos[i] + "" + [j+43] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+44] + "=" + analitos[i] + "" + [j+44] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+45] + "=" + analitos[i] + "" + [j+45] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+46] + "=" + analitos[i] + "" + [j+46] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+47] + "=" + analitos[i] + "" + [j+47] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+48] + "=" + analitos[i] + "" + [j+48] + "/COUNT'+
" ' "+ analitos[i] + "" + [j+49] + "=" + analitos[i] + "" + [j+49] + "/COUNT'+

```

*//Calcula média das realizações simuladas dentro de cada painel k-mean*

```

" 'SUM=0'+
" 'SUM1=" + analitos[i] + "" + [j] + "+" + analitos[i] + "" + [j+1] + "+" + analitos[i]+""+
"" + [j+2] + "+" + analitos[i] + "" + [j+3] + "+SUM'+
" 'SUM2=" + analitos[i] + "" + [j+4] + "+" + analitos[i] + "" + [j+5] + "+" + analitos[i]+""+
"" + [j+6] + "+" + analitos[i] + "" + [j+7] + "+SUM1'+
" 'SUM3=" + analitos[i] + "" + [j+8] + "+" + analitos[i] + "" + [j+9] + "+" + analitos[i]+""+
"" + [j+10] + "+" + analitos[i] + "" + [j+11] + "+SUM2'+
" 'SUM4=" + analitos[i] + "" + [j+12] + "+" + analitos[i] + "" + [j+13] + "+" + analitos[i]+""+
" + [j+14] + "+" + analitos[i] + "" + [j+15] + "+SUM3'+
" 'SUM5=" + analitos[i] + "" + [j+16] + "+" + analitos[i] + "" + [j+17] + "+" + analitos[i]+""+
" + [j+18] + "+" + analitos[i] + "" + [j+19] + "+SUM4'+
" 'SUM6=" + analitos[i] + "" + [j+20] + "+" + analitos[i] + "" + [j+21] + "+" + analitos[i]+""+
" + [j+22] + "+" + analitos[i] + "" + [j+23] + "+SUM5'+
" 'SUM7=" + analitos[i] + "" + [j+24] + "+" + analitos[i] + "" + [j+25] + "+" + analitos[i]+""+

```

```
" + [j+26] + "+" + analitos[i] + "" + [j+27] + "+SUM6'" +
" 'SUM8=" + analitos[i] + "" + [j+28] + "+" + analitos[i] + "" + [j+29] + "+" + analitos[i]+"" +
" + [j+30] + "+" + analitos[i] + "" + [j+31] + "+SUM7'" +
" 'SUM9=" + analitos[i] + "" + [j+32] + "+" + analitos[i] + "" + [j+33] + "+" + analitos[i]+"" +
" + [j+34] + "+" + analitos[i] + "" + [j+35] + "+SUM8'" +
" 'SUM10=" + analitos[i] + "" + [j+36] + "+" + analitos[i] + "" + [j+37] + "+" + analitos[i]+"" +
" + [j+38] + "+" + analitos[i] + "" + [j+39] + "+SUM9'" +
" 'SUM11=" + analitos[i] + "" + [j+40] + "+" + analitos[i] + "" + [j+41] + "+" + analitos[i]+"" +
" + [j+42] + "+" + analitos[i] + "" + [j+43] + "+SUM10'" +
" 'SUM12=" + analitos[i] + "" + [j+44] + "+" + analitos[i] + "" + [j+45] + "+" + analitos[i]+"" +
" + [j+46] + "+" + analitos[i] + "" + [j+47] + "+SUM11'" +
" 'SUM=" + analitos[i] + "" + [j+48] + "+" + analitos[i] + "" + [j+49] + "+SUM12'" +
```

```
" 'AV" + analitos[i] + "=SUM/( " + nsim + " )'" +
```

```
//Deleta campos desnecessários
```

```
" 'erase(SUM, SUM1, SUM2, SUM3, SUM4, SUM5, SUM6)'" +
" 'erase(SUM7, SUM8, SUM9, SUM10, SUM11, SUM12)'" +
```

```
//Calcula desvio padrão e coeficiente de variação
```

```
" 'SUM_SD1=0'" +
" 'SUM_SD2=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+0] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+1] + " ),2)+SUM_SD1'" +
" 'SUM_SD3=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+2] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+3] + " ),2)+SUM_SD2'" +
" 'SUM_SD4=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+4] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+5] + " ),2)+SUM_SD3'" +
" 'SUM_SD5=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+6] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+7] + " ),2)+SUM_SD4'" +
" 'SUM_SD6=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+8] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+9] + " ),2)+SUM_SD5'" +
" 'SUM_SD7=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+10] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+11] + " ),2)+SUM_SD6'" +
" 'SUM_SD8=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+12] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+13] + " ),2)+SUM_SD7'" +
" 'SUM_SD9=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+14] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+15] + " ),2)+SUM_SD8'" +
" 'SUM_SD10=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+16] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+17] + " ),2)+SUM_SD9'" +
" 'SUM_SD11=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+18] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+19] + " ),2)+SUM_SD10'" +
" 'SUM_SD12=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+20] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+21] + " ),2)+SUM_SD11'" +
" 'SUM_SD13=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+22] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+23] + " ),2)+SUM_SD12'" +
" 'SUM_SD14=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+24] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+25] + " ),2)+SUM_SD13'" +
" 'SUM_SD15=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+26] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+27] + " ),2)+SUM_SD14'" +
" 'SUM_SD16=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+28] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+29] + " ),2)+SUM_SD15'" +
" 'SUM_SD17=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+30] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+31] + " ),2)+SUM_SD16'" +
" 'SUM_SD18=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+32] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+33] + " ),2)+SUM_SD17'" +
" 'SUM_SD19=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+34] + " ),2)+pow( (AV"+
" + analitos[i] + "-" + analitos[i] + "" + [j+35] + " ),2)+SUM_SD18'" +
" 'SUM_SD20=pow( (AV" + analitos[i] + "-" + analitos[i] + "" + [j+36] + " ),2)+pow( (AV"+
```



```

"" + analitos[i] + "-" + analitos[i] + "" + [j+37] + "),2)+SUM_SD19'" +
" 'SUM_SD21=pow((AV" + analitos[i] + "-" + analitos[i] + "" + [j+38] + "),2)+pow((AV"+
"" + analitos[i] + "-" + analitos[i] + "" + [j+39] + "),2)+SUM_SD20'" +
" 'SUM_SD22=pow((AV" + analitos[i] + "-" + analitos[i] + "" + [j+40] + "),2)+pow((AV"+
"" + analitos[i] + "-" + analitos[i] + "" + [j+41] + "),2)+SUM_SD21'" +

//Deleta campos desnecessários (para respeitar o máximo de colunas permitido = 256)
" 'erase(SUM_SD1, SUM_SD2, SUM_SD3, SUM_SD4, SUM_SD5)'" +
" 'erase(SUM_SD6, SUM_SD7, SUM_SD8, SUM_SD9, SUM_SD10, SUM_SD11)'" +
" 'erase(SUM_SD12, SUM_SD13, SUM_SD14, SUM_SD15, SUM_SD16)'" +
" 'erase(SUM_SD17, SUM_SD18, SUM_SD19, SUM_SD20, SUM_SD21)'" +

" 'SUM_SD23=pow((AV" + analitos[i] + "-" + analitos[i] + "" + [j+42] + "),2)+pow((AV"+
"" + analitos[i] + "-" + analitos[i] + "" + [j+43] + "),2)+SUM_SD22'" +
" 'SUM_SD24=pow((AV" + analitos[i] + "-" + analitos[i] + "" + [j+44] + "),2)+pow((AV"+
"" + analitos[i] + "-" + analitos[i] + "" + [j+45] + "),2)+SUM_SD23'" +
" 'SUM_SD25=pow((AV" + analitos[i] + "-" + analitos[i] + "" + [j+46] + "),2)+pow((AV"+
"" + analitos[i] + "-" + analitos[i] + "" + [j+47] + "),2)+SUM_SD24'" +
" 'SUM_SD=pow((AV" + analitos[i] + "-" + analitos[i] + "" + [j+48] + "),2)+pow((AV"+
"" + analitos[i] + "-" + analitos[i] + "" + [j+49] + "),2)+SUM_SD25'" +

//Deleta campos desnecessários
" 'erase(SUM_SD22, SUM_SD23, SUM_SD24, SUM_SD25)'" +

" 'SD" + analitos[i] + "=SQRT(SUM_SD/" + nsim + ")'" +
" 'CV" + analitos[i] + "=SD" + analitos[i] + "/AV" + analitos[i] + "'" +

//Deleta campos desnecessários
" 'erase(SUM_SD)'" +

//Atribui score com base no CV
" 'S" + [k] + "ana" + [i] + "=0'" +
" 'IF (CV" + analitos[i] + "<" + threshold + ") S" + [k] + "ana" + [i] + "=1' 'END'" +
" 'GO'");

oDmApp.ParseCommand("copy &IN=temp5 &OUT=temp4");

}

var i = 0;

oDmApp.ParseCommand("selcop &IN=temp4 &OUT=temp6 *F1=kmean"+[k]+" *F2=S"+[k]+"ana"+[i]+" "+
" *F3=S"+[k]+"ana"+[i+1]+" *F4=S"+[k]+"ana"+[i+2]+" *F5=S"+[k]+"ana"+[i+3]+" @KEEPALL=1");

//Junta resultados

oDmApp.ParseCommand("mgsort &IN=temp &OUT=temp7a *KEY1=IJK @ORDER=1");

oDmApp.ParseCommand("join &IN1=temp7a &IN2=temp_loop &OUT=temp7b *KEY1=IJK @SUBSETR=0"+
" @SUBSETF=0 @CARTJOIN=0 @PRINT=0");

oDmApp.ParseCommand("mgsort &IN=temp7b &OUT=temp7 *KEY1=kmean" + [k] + " @ORDER=1");

oDmApp.ParseCommand("mgsort &IN=temp6 &OUT=temp8 *KEY1=kmean" + [k] + " @ORDER=1");

oDmApp.ParseCommand("join &IN1=temp7 &IN2=temp8 &OUT=temp9 *KEY1=kmean" + [k] + "" +
" @SUBSETR=0 @SUBSETF=0 @CARTJOIN=0 @PRINT=0");

oDmApp.ParseCommand("seldel &IN=temp9 &OUT=temp10 *F1=kmean" + [k] + " @KEEPALL=0");

```

```

oDmApp.ParseCommand("mgsort &IN=temp10 &OUT=temp_loop *KEY1=IJK @ORDER=1");

}

//Calcula CATEG

var i = 0;

oDmApp.ParseCommand("extra &IN=temp_loop &OUT=temp11 @APPROX=0 @PRINT=0 'COUNT_S=0' 'GO'");

for (var k = 1; k <= nkmean; k++) {

oDmApp.ParseCommand("extra &IN=temp11 &OUT=temp12 @APPROX=0 @PRINT=0"+
" 'COUNT_S=S"+[k]+"ana"+[i]+"S"+[k]+"ana"+[i+1]+"S"+[k]+"ana"+[i+2]+" "+
" S"+[k]+"ana"+[i+3]+"COUNT_S'+"+
" 'GO'");

oDmApp.ParseCommand("copy &IN=temp12 &OUT=temp11");

}

oDmApp.ParseCommand("extra &IN=temp11 &OUT=temp13 @APPROX=0 @PRINT=0"+
"'CATEG=2'+
"'RATE=COUNT_S/(" + analitos.length + "*" + nkmean + ")'+
"'IF (RATE>" + rate + ") CATEG=1 END'+
"'GO'");

//Adiciona CATEG ao modelo total

oDmApp.ParseCommand("selcop &IN=temp13 &OUT=temp14 *F1=IJK *F2=CATEG @KEEPALL=1");

oDmApp.ParseCommand("MGSORT &IN=temp14 &OUT=temp15 *KEY1=IJK @ORDER=1.0");

oDmApp.ParseCommand("join &IN1=" + mdreg + " &IN2=temp15 &OUT=temp16 *KEY1=IJK"+
" @SUBSETR=0 @SUBSETF=0 @CARTJOIN=0 @PRINT=0");

oDmApp.ParseCommand("MGSORT &IN=temp16 &OUT=" + mdout + " *KEY1=IJK @ORDER=1.0");

//Deleta temporários
oDmApp.ActiveProject.Deletefile('temp*');

    alert("PROCESSO CONCLUÍDO!");

}

```

```

// =====
// Combina Categ referente aos dois períodos (duas e quatro semanas) para gerar CATEG 1, 2 e 3
// =====

function btnExecute_CombinaCateg() {

//Entradas
var mdin1 = "md5sim_categ_kmean40" // Arquivo de modelo categorizado por quatro semanas
var mdin2 = "md5sim_categ_kmean70" // Arquivo de modelo com categorizado por duas semanas

//Saídas
var mdout = "md5sim_categf" // Arquivo de modelo final com categorias

//=====

oDmApp.ParseCommand("extra &IN="+mdin1+" &OUT=temp @APPROX=0 @PRINT=0'CATEGP1=CATEG'+
" 'erase(CATEG) ' 'GO'");

oDmApp.ParseCommand("addmod &IN1="+mdin2+" &IN2=temp &OUT=temp1");

oDmApp.ParseCommand("extra &IN=temp1 &OUT=temp2 @APPROX=0 @PRINT=0"+
"'IF (CATEG==1) CATEGF=1 END'+
"'IF (CATEG==2) CATEGF=2 END'+
"'IF (CATEGP1==2 AND CATEG==2) CATEGF=3 END'+
"'erase(CATEGP1,CATEG)'+
"'GO'");

oDmApp.ParseCommand("MGSORT &IN=temp2 &OUT=" + mdout + " *KEY1=IJK @ORDER=1.0");

//Deleta temporários
oDmApp.ActiveProject.Deletefile('temp*');

    alert("PROCESSO CONCLUÍDO!");

}

</SCRIPT>
<body bgcolor="RGB(71,89,163)" language="javascript" onload="AutoConnect()" onunload="top.DCM =
null; top.dmBrowser = null; dm = null;">
<table style="width: 200px" border="1" cellpadding="2" cellspacing="2">
<tr>
<td align="center" colspan="2">

</td>
</tr>
<tr>
<td align="center"> PPMT<BR> SIMULATION
<p><input style="width: 170px" type="button" name="btnExecute" value="CATEG CV"
onclick="return btnExecute_CategCoV()"/></p>
<p><input style="width: 170px" type="button" name="btnExecute" value="Combina CATEG"
onclick="return btnExecute_CombinaCateg()"/></p>
<p></p>
</td>
</tr>
</table>
</body>
</html>

```