

Universidade de São Paulo  
Instituto de Geociências

GRADIENT BOOSTING  
PARA A CLASSIFICAÇÃO DE FÁCIES UTILIZANDO PERFIS DE  
POÇOS

Relatório Final

TF 2020/04

Aluno: Candido Matos Junior  
Orientador: Prof. Dr. Roberto Hirata Junior

São Paulo  
2020

CANDIDO MATOS JUNIOR

GRADIENT BOOSTING PARA A CLASSIFICAÇÃO  
DE FÁCIES UTILIZANDO PERFIS DE POÇOS

Trabalho de Formação  
apresentado como requisito parcial  
para obtenção de grau de  
Bacharel em Geologia pela  
Universidade de São Paulo.

Orientador: Prof. Dr. Roberto  
Hirata Junior

São Paulo  
2020

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Serviço de Biblioteca e Documentação do IGc/USP. Ficha catalográfica gerada automaticamente com dados fornecidos pelo(a) autor(a) via programa desenvolvido pela Seção Técnica de Informática do ICMC/USP  
Bibliotecários responsáveis pela estrutura de catalogação da publicação: Sonia Regina Yole Guerra - CRB-8/4208 | Anderson de Santana - CRB-8/6658

Matos Junior, Candido

Gradient Boosting para a classificação de fácies utilizando perfis de poços. / Candido Matos Junior - São Paulo, 2020.

48 p.: il.

Trabalho de formatura : IGc/USP  
Orient.: Hirata Junior, Roberto

1. Aprendizado de máquina 2. Classificação de fácies 3. Gradient Boosting I. Título

*Então Samuel pegou uma pedra e a ergueu entre Mispá e Sem;  
e deu-lhe o nome de Ebenézer, dizendo: "Até aqui o Senhor nos ajudou".*

1 Samuel 7:12

## AGRADECIMENTOS

Depois de um ano difícil e tecendo a parte final deste projeto com uma fratura no cotovelo após um acidente, consegui finalizá-lo. Agradeço primeiramente a Deus, sem a qual não estaria realizando tudo isso. A minha família, minha mãe, Maria, pai, Cândido Matos, minha irmã Naiara, meus padrinhos Rodolfo e Geni que me deram estada e carinho enquanto realizava o curso, sem a qual não teria condições de realizá-lo, meu primo Klaus Stockl e a minha namorada, parceira, Lidiana, que me ajudou muito durante o curso.

A todos os meus docentes do Instituto de Geociências, a todos alunos que me ajudaram: Adriana Virmond, Eduardo Julião, Erílio Santos, Gabriel Soares, Victor Santos, Geovana Geraldo.

Agradeço ao professor Roberto Hirata Jr. do IME-USP, que me deu oportunidade antes mesmo de conhecê-lo e confiou muito em mim.

E dedico este trabalho a meus avós Derci e Elidia Alves que faleceram em outubro deste ano.

## RESUMO

Aprendizado de Máquina ou “Machine Learning” é uma ferramenta bastante utilizada no momento, por estatísticos e cientistas de dados para ajudar a análise de um número elevado de dados, além de pertencer a uma subárea do campo da Inteligência Artificial, que consiste em grande parte em fazer previsões por reconhecimento de padrões nos dados de atributos ou *features* através do uso de métodos de aprendizado estatístico e computacional. Geocientistas estão utilizando Aprendizado de Máquina combinado aos diferentes processos e métodos de interpretação e uma das aplicações é a classificação de fácies, sendo crucial para interpretação de perfis de poços, visto que rochas diferentes têm permeabilidade e saturação de fluido distintos para uma dada porosidade.

A finalidade deste trabalho, com levantamento bibliográfico, é mostrar esta ferramenta computacional como um mecanismo agregador ao desenvolvimento da área de geociências, discutindo a forma como ela pode atuar desde a análise de dados até a interpretação de resultados no reconhecimento de padrões, mostrar a relação de comportamento de um conceito geológico descritivo interpretativo, como fácies, com a performance de um algoritmo de Aprendizado de Máquina, a verificar quanto às técnicas computacionais podem auxiliar o conhecimento geológico.

O estudo pretende mostrar a importância do tratamento dos dados brutos e seus métodos aplicados, bem como os processos necessários para definir um modelo preditivo de classificação de fácies. O conjunto de dados de perfis de poços deste localizados no sudoeste do estado do Kansas, nos Estados Unidos, denominado Panoma Field. O processamento dos dados brutos e aplicação de *feature engineering* melhoraram os resultados de acurácia e o algoritmo Gradient Boosting desempenhou de forma satisfatória, sempre verificando os métodos de otimização dos hiperparâmetros para produzir melhores resultados. A aplicação do modelo otimizado aos dados de teste obteve poços preditivos muito próximos aos dados reais, alcançando uma acurácia de 0,63, e quando se considera as fácies adjacentes este valor chega a 0,93.

**Palavras-chave:** Aprendizado de Máquina, classificação de fácies, Gradient Boosting

## **ABSTRACT**

Machine Learning is a tool widely used at the moment by statisticians and data scientists to help analyze a large number of data, in addition to belonging to a subarea of the Artificial Intelligence field which consists of largely making prediction by pattern recognition in the features through the use of statistical learning methods.

Geoscientists are using machine learning combined with the different processes and methods of interpretation. Besides that, one of the applications is the facies classification, Geoscientists are using Machine Learning combined with the different processes and methods of interpretation and one of the applications is the facies classification, being crucial for the interpretation of well logs, since different rocks have different permeability and fluid saturation for a given porosity.

The purpose of this work, with bibliographic survey, is to show this computational tool as an aggregating mechanism for the development of the geosciences area, discussing how it can act from data analysis to the interpretation of results in pattern recognition, showing the relationship of behavior of a descriptive interpretive geological concept, such as facies, with the performance of a Machine Learning algorithm, to verify how computational techniques can help geological knowledge.

The study aims to show the importance of processing raw data and its applied methods, as well as the processes necessary to define a predictive model for facies classification. The data set of well logs from is located in southwest Kansas, in the United States, called Panoma Field. The processing of raw data and the application of feature engineering improved the accuracy results and the Gradient Boosting algorithm performed satisfactorily, always checking the optimization methods of the hyperparameters to produce better results. The application of the optimized model to the test data obtained predictive wells very close to the real data, reaching an accuracy of 0.63, and when considering the adjacent facies this value reaches 0.93.

**Keywords:** Machine Learning, facies classification, Gradient Boosting

## **SUMÁRIO**

1. INTRODUÇÃO	7
2. OBJETIVOS E MEIOS	8
3. TRABALHOS DE REFERÊNCIAS	8
3.1. Área de estudo	8
3.2. Contexto geológico	10
3.3. Perfis geofísico e as variáveis do problema	12
3.4. Feature Engineering	15
3.5. Aprendizado de Máquina	18
3.6. Algoritmos	21
3.6.1. Métodos Baseados em Árvores	22
3.6.1.2. Árvores de Decisão	22
3.6.1.3. Florestas Aleatórias	22
3.6.1.4. Gradient Boosting	23
3.7. Validação e Avaliação	25
3.8. Métrica de classificação	26
3.9. Ajuste do modelo	27
4. RESULTADOS OBTIDOS	28
5. INTERPRETAÇÃO E DISCUSSÃO DOS RESULTADOS	43
6. CONCLUSÃO	46
7. REFERÊNCIAS	47

## 1. INTRODUÇÃO

A formação do pesquisador da área de geociências nos dias de hoje pode fazer uso de recursos na área de Ciência de Dados. E um dos métodos mais importantes de análise de dados é o Aprendizado de Máquina.

Aprendizado de Máquina ou *Machine Learning* é uma ferramenta muito utilizada no momento, por estatísticos e cientistas de dados para ajudar a análise de um número elevado de informações, além de pertencer a uma subárea do campo da Inteligência Artificial, que consiste em grande parte em fazer previsões por reconhecimento de padrões nos dados de atributos ou *features* através do uso de métodos de aprendizado estatístico (Hastie et al., 2001).

Geocientistas estão utilizando Aprendizado de Máquina combinado aos diferentes processos e métodos de interpretação e uma das aplicações é a classificação de fácies utilizando dados de perfis de poços (Hall, 2016; Bestagini, 2017; Guarido, 2018). O conjunto de dados explorado neste estudo é oriundo da Universidade de Kansas, nos Estados Unidos (*Kansas University, USA*) de seu departamento de pesquisa geológico, *Kansas Geological Survey* (KGS), que tem como estudo o campo de gás Hugoton e Panoma, sudoeste do Kansas (Figura 1). Dubois et al. (2007) utilizam este conjunto de exemplos como dados de treino e de teste para classificação de fácies, observando o comportamento de cada classificador.

O tipo de dado que fornecerá as informações para o desenvolvimento do projeto se dá por meio de perfis geofísicos de poços, os denominados *well logs*, utilizados pela indústria de petróleo e gás. As medidas dos perfis de poços, os dados de entrada, são: raios gama (GR), resistividade (ILD\_log10), efeito fotoelétrico (PE), diferença de porosidade de densidade de nêutrons e porosidade de densidade de nêutrons média (DeltaPHI e PHIND); e as nove amostras de rochas ou fácies correspondentes a estes perfis de poços, através da descrição de aproximadamente 609 metros de testemunho e análise de perfis de poços (Dubois, et al., 2003 a,b).

A linguagem usada é Python, versão 3.9.0, *open source*, não sendo necessária nenhuma licença para uso e suas plataformas de programação Jupyter e Google Collaboratory. O algoritmo treina o conjunto de dados, fácies (*labels*) associadas às suas correspondentes medidas de *well logs*, fazendo com que se aplique a estes dados um modelo de aprendizado, e posteriormente, aplicada uma validação, submeter a um dado conjunto de poços que não possui dados de fácies (*output*). Criando, assim, um modelo preditivo de fácies baseado no processo de aprendizado (Bishop, 2006; Hall, 2016), a fim de garantir a automatização, reproduzibilidade e o reconhecimento de padrões do processo que o procedimento usual não possui.

## 2. OBJETIVOS E MEIOS

A finalidade deste trabalho é mostrar esta ferramenta computacional como um mecanismo agregador ao desenvolvimento da área de geociências, discutindo a forma como ela pode atuar, desde a análise de dados até a interpretação de resultados no reconhecimento de padrões, mostrar a relação de comportamento de um conceito geológico descritivo interpretativo, como fácies, com a performance de um algoritmo de Aprendizado de Máquina. Ademais, o estudo pretende mostrar a importância do tratamento dos dados brutos e seus métodos aplicados bem como os processos necessários para definir um modelo preditivo de classificação de fácies.

Para isso, foi realizado um levantamento bibliográfico de outros trabalhos que usaram Aprendizado de Máquina na área de geociências para classificação de fácies utilizando perfis de poços, discutindo métodos utilizados no processamento dos dados bem como executar ajustes que possam melhorar o modelo, além da busca de entendimento do funcionamento da perfilagem geofísica de poços com seus parâmetros medidos para desenvolver melhor a pesquisa. Desse modo, a implicação do estudo é uma manipulação dos parâmetros dos dados, as propriedades físicas das rochas, tais como porosidade, densidade, resistividade, radioatividade, bem como a manipulação das bibliotecas Numpy, Pandas, Matplotlib, Seaborn e Sklearn da linguagem Python.

## 3. TRABALHOS DE REFERÊNCIAS

### 3.1. Área de estudo

Dubois et al. (2003 a,b) utilizam estes parâmetros como dados de treino e de teste para classificação de fácies e observa-se o comportamento de cada classificador, dados localizados no campo de gás Hugoton e Panoma, no sudoeste do Kansas (Figura 1).

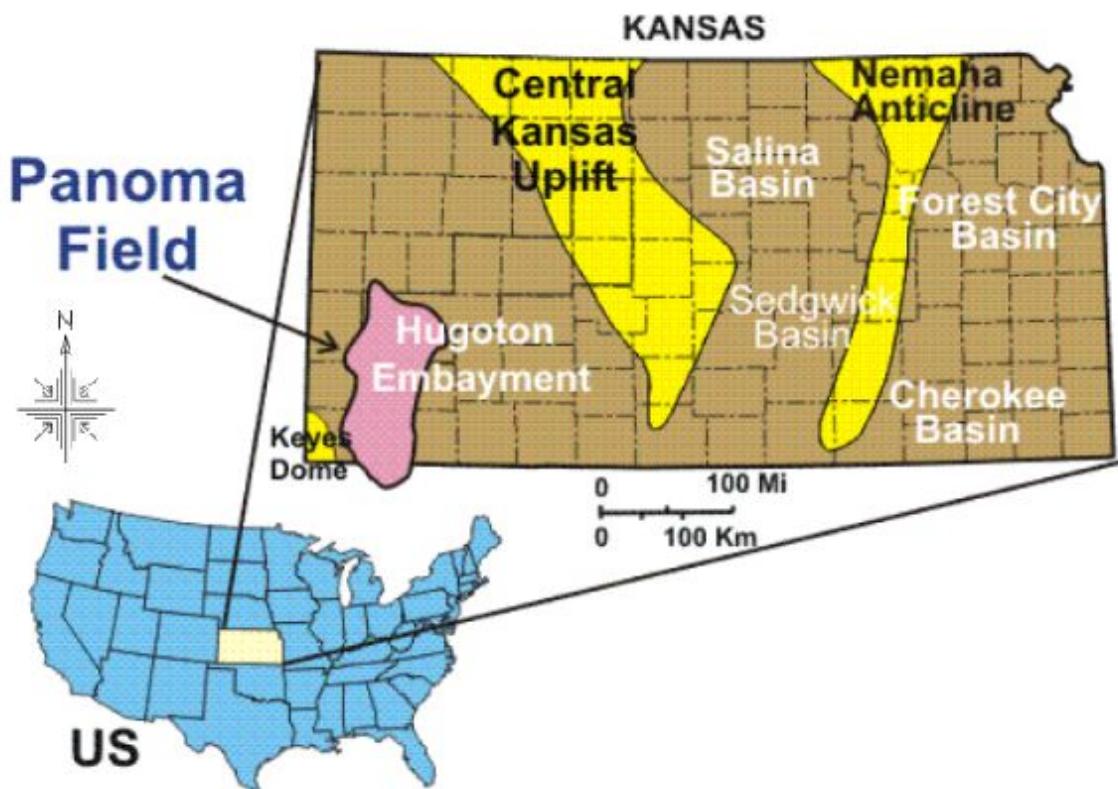


Figura 1: Campo de Panoma é localizado no Embasamento Hugoton da Bacia Anadarko no sudoeste do estado de Kansas- Estados Unidos (Modificado de Bohling & Dubois, 2003).

O Grupo Permiano Council Grove no Campo de Panoma, sudoeste do estado de Kansas, produziu 80 x 109 metros cúbicos de gás de aproximadamente 2600 poços de um intervalo de 60 metros em profundidades de 800 a 1.000 metros desde sua descoberta na década de 1960. A saturação inicial de gás, as taxas de produção e a produção acumulada no Campo de Panoma são controladas pela distribuição de porosidade e permeabilidade no campo, que por sua vez são controladas pela distribuição de fácies ( Dubois et al., 2003 a, b).

Esse conjunto de dados é de 10 poços: SHRIMPLIN, SHANKLE, LUKE G U, CROSS H CATTLE, NOLAN, Recruit F9, NEWBY, CHURCHMAN BIBLE, ALEXANDER D, KIMZEY A (4149 instâncias), consistindo em um conjunto de sete variáveis preditoras e nove fácies (classe) para cada valor de entrada de exemplo e para dados de validação, dois poços de teste sem informação de classes, STUART e CRAWFORD (830 instâncias) com as mesmas variáveis preditoras (Figura 2).

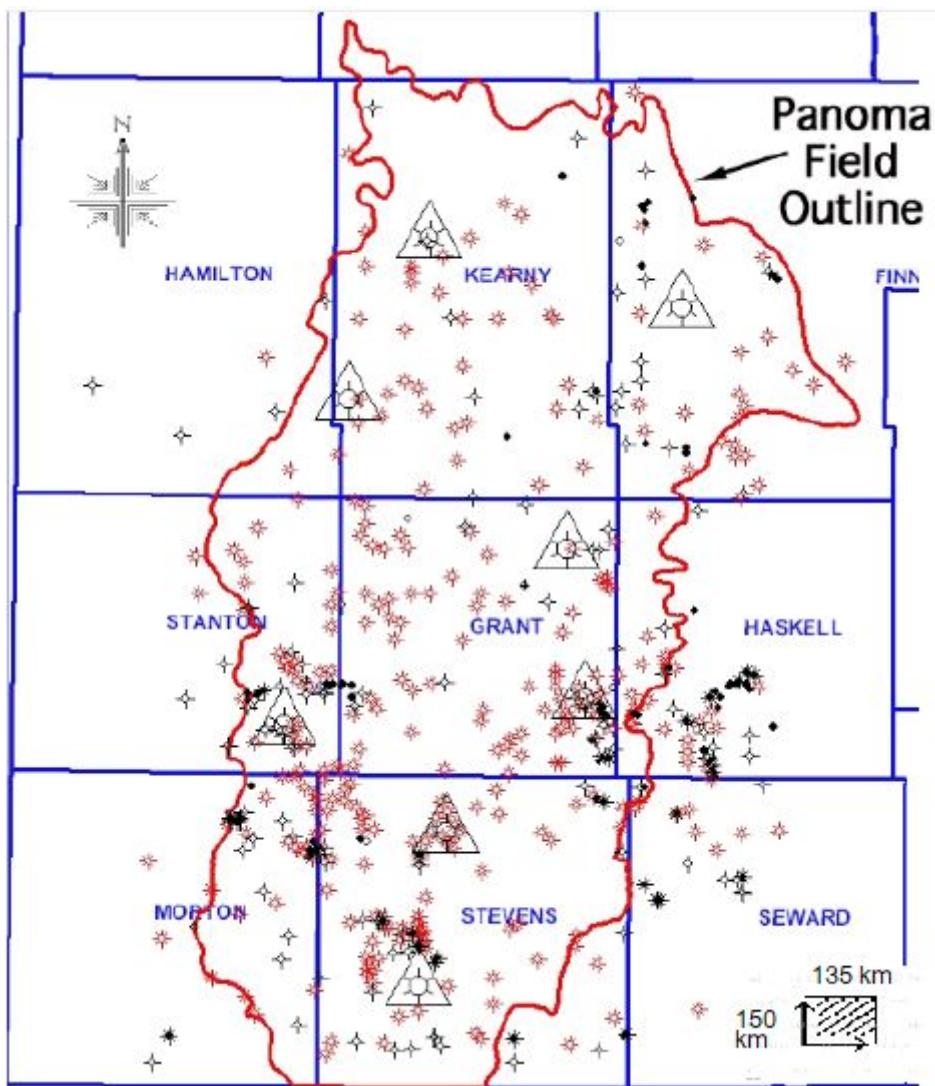


Figura 2: Mapa mostrando a localização de oito poços no conjunto de treinamento (triângulos grandes) e 515 poços sem informações de litofácies (sem triângulos). As dimensões do mapa são aproximadamente 150 por 135 km (Modificado de Dubois et al., 2005).

### 3.2. Contexto Geológico

O Council Grove Group é um grupo geológico em Kansas, Oklahoma, Nebraska, e Colorado. Este grupo preserva fósseis que datam do limite Carbonífero-Permiano, está abaixo do Grupo Chase, onde se localiza o campo Hugoton, na estratigrafia (Dubois et al., 2001; Dubois et al., 2003) observada na Figura 3.

# Stratigraphy

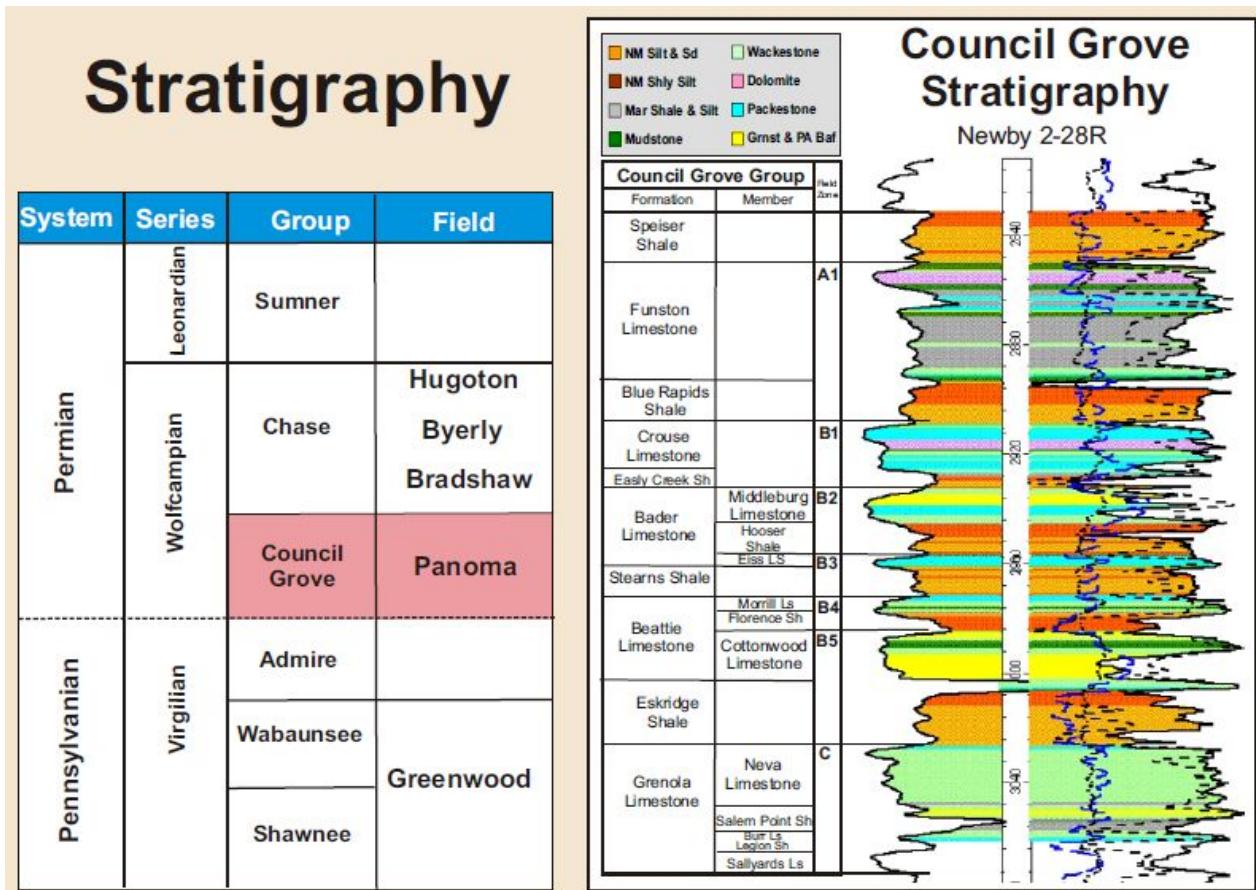


Figura 3: Coluna estratigráfica da região da área de estudo (Modificado de Dubois et al., 2003).

Grupo Council Grove é composto por sequências marinhas-não-marinhas de quarta ordem delimitadas por inconformidades na superfície de carbonato exposto.

Uma típica sucessão vertical, começando na exposta superfície de carbonato, contendo, principalmente, silte, areias muito finas e silte-argilosos ricos com paleossolos. Acima de carbonatos, estes são cobertos por águas mais profundas de siltitos marinhos escuros e carbonato, sobrepostos por grainstones, interpretados para indicar aumento da onda ou agitação das marés; também dolomitos siltosos, onde houve pouca ou nenhuma agitação das ondas (Dubois et al., 2003 a, b)- Figura 4.

# Council Grove Lithofacies



Figura 4: Imagens de 5 das 8 classes de litofácies e ambiente deposicional do grupo Council Grove (Modificado de Dubois et al., 2003b).

### 3.3. Perfilas geofísicos e as variáveis do estudo

O tipo de dado que fornece as informações para o desenvolvimento do projeto, se dá por meio de perfis geofísicos de poços, os denominados *well logs*, utilizados pela indústria de petróleo e gás. Eles têm a vantagem de medir propriedades de rochas *in situ* que não podem ser medidas em laboratório por amostras de testemunhos ou cortes de maior custo econômico (Bestagini, 2017).

Os parâmetros medidos não fornecem informações precisas de quanto gás e óleo existem na subsuperfície ou quanto serão produzidos, no entanto, destes parâmetros derivam informações importantes que podem, com pressupostos confirmados, guiar uma estimativa significativa na busca por hidrocarbonetos. Tais parâmetros são: porosidade, densidade, resistividade, radiação gama, velocidade, efeito fotoelétrico (Bjørlykke et al., 2015).

A perfilação é feita através de um cabo de aço que é deslocado pelo poço gerado (Figura 5), na extremidade deste cabo existem eletrodos que captam os sinais de medições que serão processados na estação de tratamento de dados, dados armazenados em um sistema computacional e estes são produzidos em papel ou filme (Bjørlykke et al., 2015), e atualmente por softwares.

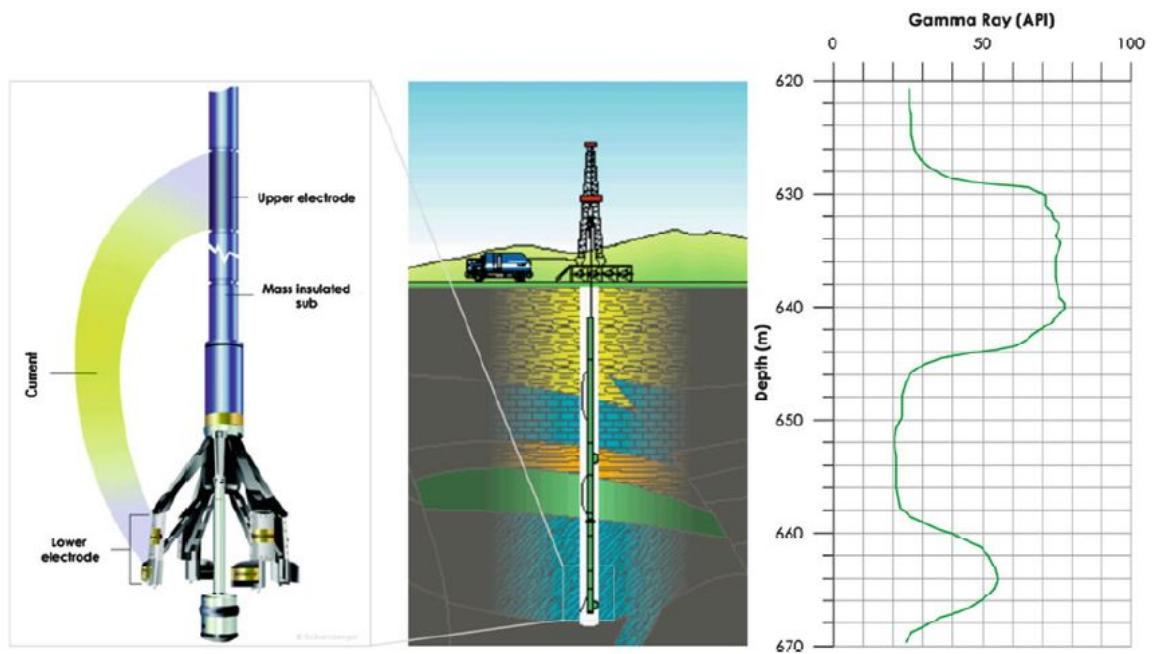


Figura 5: Exemplos de uma ferramenta de perfilagem e sua operação, ao lado direito, um perfil gerado de radiação gama (Bjørlykke et al., 2015).

Uma fundamental propriedade volumétrica de rocha, que pode ser medida pela perfilagem geofísica de maneira indireta, é a porosidade. Ela descreve o potencial volume armazenado de fluidos (água, gás e óleo) e influencia a maior parte das propriedades das rochas tais como velocidade de onda elástica, resistividade elétrica, densidade (Schön , 2016). Existe uma forte relação entre densidade e porosidade, para rochas sedimentares, há geralmente um intervalo amplo de valores de densidade que refletem variações de porosidade e saturação, de modo que valores baixos de densidade resultam do aumento da porosidade e/ou aumento de gás nos poros (Schön , 2016).

A variável radiação gama é muito utilizada em rochas reservatório, em particular arenitos e carbonatos, na medição de densidade aplicada para cálculo de porosidade, devido à maior penetração em relação à radiação alfa e beta, sendo os três principais elementos radioativos de ocorrência nas rochas são: Tório, Urânio e Potássio (Schön, 2016). Geralmente, folhelhos terão maior nível de radioatividade que outros sedimentos, consequentemente, sonda de raios gama é usada para fazer esta distinção, entretanto, carbonatos e arenitos apresentam índices baixos de radioatividade (Schön, 2016). A medida de efeito fotoelétrico é produto da interação da radiação gama e as propriedades nucleares das rochas (Schön, 2016).

A porosidade também é verificada através da interação da radiação de nêutrons, de forma que em uma rocha saturada de óleo e água, como folhelho, a resposta de nêutrons

reflete a porosidade, ao passo que uma rocha com presença de gás, de baixo teor de hidrogênio, reflete um efeito de nêutrons menor (Schön, 2016).

A resistividade, propriedade elétrica da rocha, indica fração de volume da rocha (porosidade, saturação) e distribuição do eletrólito na rocha, de modo que poros saturados de água possuem resistividade baixa com aumento da porosidade e fraturamento, no entanto, poros com óleo e gás apresentam resistividade maior (Schön, 2016).

Para este trabalho, as fácies são baseadas na descrição de aproximadamente 609 metros de testemunho a intervalos de amostragem de 15 cm (Dubois et al., 2003b; Dubois et al., 2005). As variáveis são essencialmente contínuas são raios gama (GR), resistividade (ILD\_log10), efeito fotoelétrico (PE), diferença de porosidade de densidade de nêutrons e porosidade de densidade de nêutrons média (DeltaPHI e PHIND). As nove fácies (classes de rochas) são: arenito não marinho (SS), siltito grosso não marinho (CSiS), siltito fino não marinho (FSiS), siltito e xisto marinhos (SiSh), mudstone (calcário) (MS), wackestone (WS) (calcário), dolomito (D), packstone-grainstone (PS) (calcário), bafflestone filóide-algal (calcário) (BS). As fácies são discretas que possuem fácies vizinhas bastante próximas, expostas na Tabela 1.

Tabela 1: As fácies e seus nomes abreviados e seus vizinhos próximos.

Fácies	Siglas	Fácies Adjacentes
1	SS	2
2	CSiS	1,3
3	FSiS	2
4	SiSh	5
5	MS	4,6
6	WS	5,7
7	D	6,8
8	PS	6,7,9
9	BS	7,8

### 3.4. Feature Engineering

A sistematização científica que foi adotada leva em consideração a questão geológica que se pretende responder, no caso, é um modelo preditivo de fácies que automatize por padrões o processo de análise dos perfis geofísicos no âmbito de exploração de óleo e gás. Considerar um conjunto de atributos, variáveis ou o termo em inglês, que será usado neste trabalho, *features*; a terminologia para isso é “processamento de *features*”, que é a verificação de *features* usados para o problema, seleção de um modelo final confiável e, por fim, fazer a classificação que seleciona por aprendizado seus dados no espaço de entrada automaticamente e os resultados da predição são analisados (Smith, 2018).

Em Aprendizado de Máquina, *features* descrevem aspectos dos dados de determinado problema, além de distinguir e caracterizar diferentes grupos de objetos. O seu tratamento e verificação é de suma importância para gerar modelos acurados de predição (Guozhu & Huan, 2018). Além disso, estes componentes devem derivar de maneira natural do tipo de dado bruto que está disponível (Zheng & Casari, 2018), não somente se os dados contêm informações importantes, mas também qual informação existe, através de regras e *features* descobertos (Liu & Motoda, 1998). A relação *features* e modelos é muito estreita visto que alguns modelos são mais apropriados para alguns tipos de variáveis e vice-versa, isso porque determinados *features* devem ser mais fáceis para a modelagem assimilar (Zheng & Casari, 2018). E seus tipos são classificados como: binário, categórico, discreto, contínuo e complexo (Duboue, 2020).

O processo de tratamento para formulação de *features* mais adequado aos dados, modelo e a tarefa, é conhecido por *feature engineering* (Figura 6), esta etapa é melhor compreendida após a execução da análise exploratória de dados brutos que extrai informações sobre o comportamento dos dados, principalmente, referente a determinado *feature* analisado e uma grande quantidade de dados pode auxiliar a capacidade de retirar *features* significativos (Duboue, 2020).

Análise exploratória é um importante primeiro passo para analisar a variedade de valores dos dados brutos dispostos em diferentes colunas, fazendo uso de análise estatística descritiva tais como média, mediana, moda, extremos (máximo e mínimo), variância, desvio padrão e visualizações como box-plots são bastante úteis neste estágio (Duboue, 2020).

Muitos problemas, que se tentam resolver em Aprendizado de Máquina, não obtêm êxito devido aos parâmetros utilizados e a maneira de resolver é fazer uso de engenharia de parâmetros (“*feature engineering*”, que será abreviado daqui em diante por FE) (Domingos, 2012), este processo basicamente (1) identifica *features* considerados bons e expande-os; (2) identifica *features* redundantes e elimina-os (Duboue, 2020).

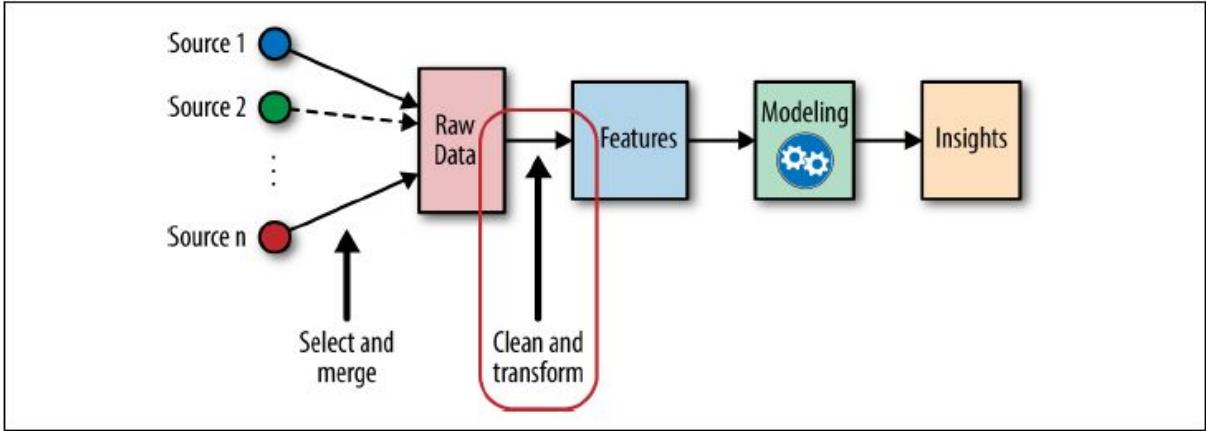


Figura 6: Local de atuação de *Feature Engineering* no fluxo de processo de ciência de dados (Zheng & Casari, 2018).

Duboue (2020) divide este processo em feature combinado, expandido e reduzido. Combinado refere-se a normalização, discretização. Normalização é uma maneira de reduzir as variações nos valores de *feature*, variações ou ordem de grandeza que muitas vezes prejudicam o modelo, certificando que esses valores sempre estão em um mesmo intervalo durante o treinamento. Alguns modelos como Máquina de Vetor Suporte (Support Vector Machine) e Redes Neurais necessitam de dados de entrada escalados a intervalos específicos como 0 e 1 ou -1 e 1, dessa forma é necessário subtrair o mínimo valor de cada variável e dividir pela diferença entre o máximo e o mínimo (Duboue, 2020). Outra maneira, é a padronização que transforma features com média zero e unidade de variância, através da subtração de cada ponto dividido pelo desvio padrão. Processo *whitening* que padroniza e também remove a correlação entre as variáveis, a linearidade entre os dados, aplicado a um domínio não linear (Duboue, 2020).

Feature expandido, refere-se à imputação de variáveis ausentes, lidar com dados ausentes é uma realidade recorrente em ciência de dados, excluir os dados que não tem informação pode ser uma alternativa quando conjunto de dados de treino é grande o suficiente, no entanto, dados ausentes em geral tendem a ser representativos de um fenômeno particular dos dados coletados, podendo, então, aumentar o erro quando existe um esforço de imputar informações aos dados faltantes, isto é, adicionar um maior viés (Duboue, 2020). A considerar o tipo de problema, uma maneira de lidar com isso é substituir com valores representativos como a média, mediana ou a moda para fazer com que o algoritmo ignore valores ausentes de *features*, logo uma solução seria checar se funciona a imputação de valores para o algoritmo e comparar o desempenho do modelo de treino nos dados com os dados imputados (Duboue, 2020). Ainda assim, a forma mais confiável de imputar dados é treinar um classificador ou um regressor de *features* restantes para prever

dados ausentes, um exemplo de escolha confiável e de rápida operação é o algoritmo K-vizinhos mais próximos-KNN (Duboue, 2020).

Feature reduzido, seleção de feature e redução de dimensão completam o processo de *feature engineering*. Seleção de features é uma forma de escolher um subconjunto de features que são úteis para construir uma ferramenta preditora (Duboue, 2020), um conjunto reduzido torna o uso computacional factível de certos algoritmos, levando-os uma qualidade melhor de resultados (Guozhu & Huan, 2018). Redução dos dados também é uma técnica que consiste em reduzir a representação dos dados em um volume de dados menor sem perder a integridade do conjunto original, uma forma de reduzir é por remoção de atributos repetitivos ou dimensões e atributos irrelevantes, uma técnica importante para reduzir dimensão, visto que à medida que a dimensionalidade aumenta, a visualização e a modelagem tornam-se cada vez mais difíceis, pois dados começam a ficar mais esparsos, uma forma, portanto, de tratar este problema é aplicar uma Análise de Principais Componentes ou Principal Components Analysis (PCA), de modo que despreze componentes insignificantes, eliminando casos de menor variância de modo a visualizar melhor os dados (Alasadi & Bhaya, 2017; Zheng & Casari, 2018; Duboue, 2020). Dessa maneira, com menos *features*, pode-se manipular mais facilmente os dados, organizando as formas mais relevantes para melhorar o aprendizado (Liu & Motoda, 1998). Outra técnica de transformação dos dados é *clustering* ou agrupamento que consiste em particionar dado conjunto de dados em *clusters* ou grupos que busquem uma similaridade entre objetos de um mesmo grupo e diferença entre os vários grupos formados, sendo a similaridade medida frequentemente através de uma função de distância (Guozhu & Huan, 2018), as bibliotecas de Aprendizado de Máquina, como Sklearn que oferece vários algoritmos de *clustering* de acordo com dados usados e um tipo muito utilizado é o algoritmo *K-means* (Guardo, 2018).

Uma técnica de *feature engineering* foi utilizada no pré-processamento dos dados, denominada *feature augmentation*, para separar melhor as classes, somando algumas não linearidades, desenvolvido por Bestagini et al (2017). Primeiramente, calcula-se o vetor *feature augmented*  $f'_{d,w}$  aplicando expansão quadrática a *feature*  $f_{d,w}$ , vetor associado à profundidade e ao poço:

$$f'_{d,w} = f_{d,w}^2 = [(f_{d,w}^{GR}) \cdot (f_{d,w}^{GR}), (f_{d,w}^{RES}) \cdot (f_{d,w}^{RES}), \dots] \quad (\text{Bestagini et al., 2017})$$

Todas as operações são aplicadas como produto de matrizes de mesma dimensão. Além disso, calcula-se vetor augmented  $f''_{d,w}$  considerando todos os termos de interação de segunda ordem:

$$f_{d,w}'' = [f_{d,w}^{GR} \cdot f_{d,w}^{RES}, f_{d,w}^{GR} \cdot f_{d,w}^{PE}, \dots, f_{d,w}^{NM} \cdot f_{d,w}^{RP}] \quad (\text{Bestagini et al., 2017})$$

De fato, considerando que as fácies nas camadas vizinhas são fortemente correlacionadas, ou seja, *features* das camadas  $d + 1$  e  $d - 1$  podem ajudar a classificar a camada  $d$ . Dessa forma, define-se o gradiente de *feature augmented*:

$$f_{d,w}''' = \frac{f_{d-1,w} - f_{d,w}}{\Delta depth} \quad (\text{Bestagini et al., 2017})$$

onde todas as operações são aplicadas como produto de matrizes de mesma dimensão, e  $\Delta depth$  é a diferença de profundidade em metros entre as camadas  $d$  e  $d - 1$ .

Uma vez que os três features augmented são calculados, concatena-se todos eles em único vetor:

$$f_{d,w} = [f_{d,w}', f_{d,w}'', f_{d,w}'''] \quad (\text{Bestagini et al., 2017})$$

É importante esta abordagem, pois dado um valor de um *feature* como raios gama, GR, que indica transição de areia para folhelho, garante a noção intuitiva de gradiente baseado nas propriedades de dados de perfis de poços, e assim, acaba fornecendo um atributo para identificar mudanças litológicas (Bestagini et al., 2017).

Portanto, é de suma importância para o cientista de dados fazer pré-processamento de dados: a preparação e transformação dos dados de forma que o espaço de entrada possa ser facilmente interpretado pelo algoritmo, limpeza dos dados, transformação e seleção (Alasadi & Bhaya, 2017), técnicas que acabam sendo a tarefa mais importante da análise de dados.

### 3.5. Aprendizado de Máquina

Geocientistas estão utilizando Aprendizado de Máquina ligado aos diferentes processos e métodos de interpretação e uma das aplicações, no momento mais usada, é a classificação de fácies (Hall, 2016; Guarido, 2018).

A forma convencional de análise de perfis geofísicos é feita de forma analítica, com ajuda de *softwares* especializados, que não são *open source*, necessitando de licença para serem usados, o que eleva o tempo de análise e o custo de operação (Bestagini, 2017).

Além disso, os interpretadores e analistas não reconhecem todos os padrões, aqueles ocultos, que os perfis fornecem (Guardo, 2018).

Com uma metodologia de aprendizado adequada, o algoritmo pode treinar o conjunto de dados, fácies (*labels*) associadas aos seus correspondentes parâmetros de *well logs* (porosidade, resistividade, radiação de gama natural, densidade, neutrônico, efeito fotoelétrico) (Hall, 2016; Bestagini, et al., 2017; Guardo, 2018). e fazer com que se aplique a estes dados um modelo de aprendizado, e posteriormente, feita a validação, aplicar a parâmetros que não possuem dados de fácies (*output* desejado), criando um modelo preditivo de fácies (Hall, 2016; Bestagini et al., 2017; Guardo, 2018) baseado no processo de aprendizado (Bishop, 2006;). Além disso, a vantagem da classificação automática é permitir a reproduzibilidade e a padronização do processo.

O termo classificação se deve ao tipo de variável que está sendo utilizada para ser prevista como valor de saída, resposta ou *outputs*, no caso, a variável dependente, fácies, como resultado é do tipo categórica ou discreta, representada por valores inteiros. Ao passo que os valores de entrada (*inputs*) são denominados de preditores, variáveis independentes, mais usualmente chamados de características ou atributos (*features*). Quando se utilizam conjuntos de valores de entrada e de saída a fim de tentar prever novos valores de saída, este tipo de aprendizado se denomina supervisionado (Hastie et al., 2001).

As condições, para as técnicas de Aprendizado de Máquina funcionarem, são: existência de um padrão, ser difícil de modelar matematicamente e possuir uma quantidade representativa de dados (Abu-Mostafa et al., 2012).

Neste trabalho, a forma como estão dispostas as fácies associadas aos *features* em um perfil é definida por um padrão a ser descoberto, respeitando a sequência estratigráfica local. Entretanto, não há um modelo matemático que consiga definir esta classificação, somente a partir dos dados treinados é criado um sistema (Abu-Mostafa et al., 2012).

A formalização matemática para o problema de aprendizado (Abu-Mostafa et al., 2012) :

-Entrada:  $x$  - valores de entrada; um vetor de dimensão d.

-Saída:  $y$  - valores de saída; a resposta.

-Função alvo:  $F : X \rightarrow Y$  ; função com domínio X de onde vem o conjunto de dados de entrada e um contradomínio Y, dados de saída. A função alvo é a função ideal de classificação de fácies, que é desconhecida, no caso.

-Dados:  $(x_1, y_1), (x_2, y_2), \dots (x_N, y_N)$ , presentes no conjunto de dados, um vetor  $x$  associado a um vetor  $y$  .

-Hipótese:  $G : X \rightarrow Y$  ; função a ser aprendida  $g$  deve aproximar bem a função desconhecida  $F$  ,  $G$  é criada para aproximar a  $F$  , este é objetivo do aprendizado.

-Conjunto de Hipóteses:  $H = \{h\} \quad G \in H$ .

Basicamente para o processo de aprendizado, o que se almeja aproximar é a função alvo, através dos exemplos de treinamento, o conjunto de dados. O objetivo é produzir a hipótese final, que é a fórmula utilizada para classificação, com  $G$  se aproximando de  $F$ . O que conecta os exemplos de treinamento com a hipótese final é o algoritmo de aprendizado, este cria a fórmula a partir de um modelo de fórmulas candidatas, formando um conjunto de hipóteses, a partir do qual, escolhe-se uma hipótese final (Abu-Mostafa et al., 2012). Deste conjunto de hipóteses  $H$ , o algoritmo de aprendizado escolhe uma hipótese  $h$ , denota-se a função escolhida por  $G$ , compondo o modelo de aprendizado. Quando existe um modelo, há também um conjunto de hipóteses, havendo um algoritmo que produzirá uma dessas hipóteses (Abu-Mostafa et al., 2012).

Ademais, é preciso adicionar os elementos da teoria da probabilidade neste modelo de aprendizado, uma vez que existe um elemento de incerteza associado ao reconhecimento de padrões, os conceitos de probabilidade fornecem estrutura de manipulação, quantificação de incertezas e formam um dos fundamentos centrais do reconhecimento de padrões (Bishop, 2006). Dessa maneira, no espaço de entrada  $X$  dos dados é aplicada uma distribuição de probabilidade  $P$  aos pontos  $x_1, \dots, x_N$  deste espaço, ou seja, existe uma probabilidade de selecionar um ponto de maneira independente. Atrelada a isso, há a medida de erro definidora de quanto  $G$  se aproxima de  $F$  em dado ponto  $x$  de teste gerado da mesma distribuição de probabilidade. Para adequar a função alvo a esta distribuição probabilística, substitui-se  $y = F(x)$  por  $P(y/x)$ , dependendo ainda de  $x$ , assim, quando  $x$  for gerado,  $y$  também será gerado de maneira probabilística, sendo par  $(x, y)$  gerado pela distribuição conjunta  $P(x)P(y/x)$  (Abu-Mostafa et al., 2012). O diagrama abaixo, retirado de notas de aula do professor Yaser Abu-Mostafa do Caltech-EUA, propõe organizar estes conceitos apresentados (Figura 7).

## The learning diagram - including noisy target

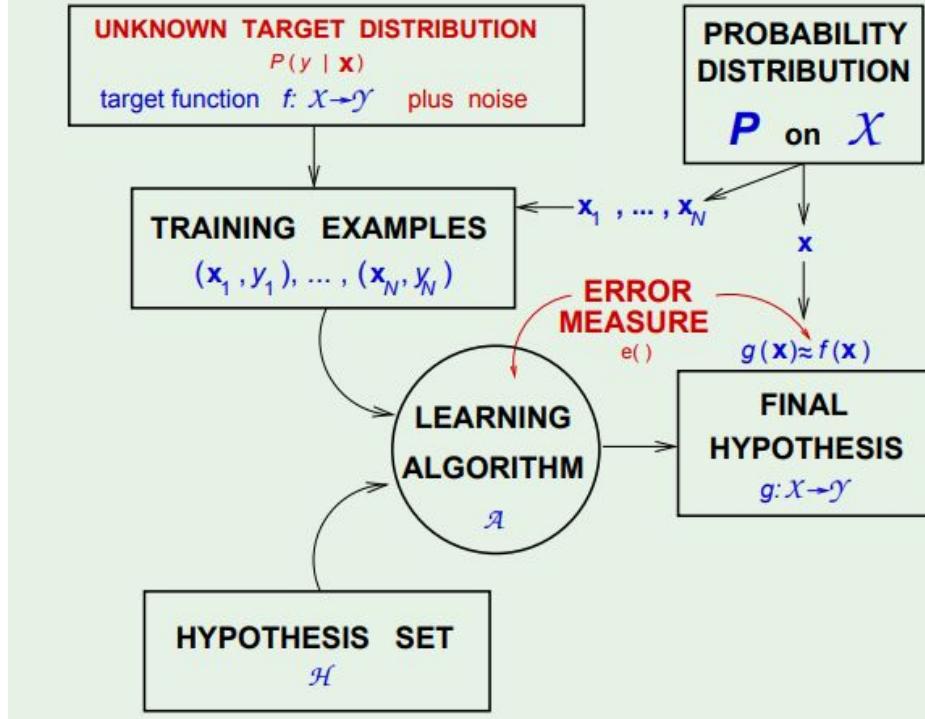


Figura 7: Elementos que compõem o diagrama de Aprendizado Supervisionado (retirado do curso *Learning from data* de Abu-Mostafa do Calltech, 2012.).

Entre os três tipos de métodos de Aprendizado de Máquina (supervisionado, não supervisionado, e reforço), o Aprendizado Supervisionado é o mais adequado para a classificação (Caté et al., 2018). Há muitos modelos de aprendizado com aplicação em geociências para classificação de fácies: Classificador Bayes (Dubois et al., 2007), K-vizinhos mais próximos, ou KNN (Caté et al., 2017), Métodos de Árvores de decisão: Gradient Boosting, Florestas aleatórias ou *Random Forest* (Hall, 2016; Caté et al., 2018; Guarido, 2017), Máquina de Vetores de Suporte, ou SVM (Caté et al., 2017; Alexsandro et al., 2017), Redes neurais (Dubois et al., 2007; Silva et al., 2014).

### 3.6. Algoritmos de Aprendizado de Máquina do problema

A metodologia para este escopo será baseada em Árvores (Breiman et al., 1984; Breiman et al., 2001) e o modelo usado para previsões é o Gradient Boosting que é uma combinação de modelos mais simples de Árvores de Decisão (Breiman et al., 1984; Breiman et al., 2001). Nesta seção, é feita a discussão sobre os métodos de Árvores e são introduzidas as definições dos seus principais tipos de métodos: Árvores de Decisão, Florestas Aleatórias, que pode ser implementado para melhorar a classificação por Árvores de Decisão, e por último, o classificador Gradient Boosting.

### **3.6.1. Métodos Baseados em Árvores**

Algoritmos baseados em Árvores são um dos mais usados em Aprendizado Supervisionado, possuem um desempenho muito satisfatório em problemas não lineares, diferentemente de quando se usa modelos lineares. Além disso, são adaptáveis para qualquer tipo de problema, tanto regressão como classificação (Breiman et al., 1984; Breiman et al., 2001). Árvores de Decisão, Florestas Aleatórias, Gradient Boosting são usados em diversos problemas em ciência de dados, e também na área de geociências (Guardo, 2018).

#### **3.6.1.2. Árvores de Decisão**

Árvores de Decisão é um tipo de algoritmo de Aprendizado Supervisionado amplamente usado em problemas de classificação. Nessas estruturas em árvores, as folhas ou nó terminal representam classes ou uma distribuição de probabilidade ligada por um conjunto de variáveis, denominado de nó de decisão (Breiman et al., 1984). A forma de criar uma Árvore de Decisão é feita por escolha de uma variável a cada passo que melhor divide um conjunto de itens (Breiman et al., 1984). Utilizam-se diferentes métricas, para conseguir melhores divisões em cada nó que fornecem a homogeneidade da variável alvo dentro dos subconjuntos de forma que os valores resultantes são combinados para fornecer a medida da qualidade da divisão do nó em dois ou mais subnós (Breiman et al., 1984), uma das métricas mais usadas é impureza Gini (Breiman et al., 1984).

Árvore de Decisão está distante de ser uma ferramenta ideal para o aprendizado preditivo, devido à sua imprecisão (Breiman et al., 1984), isto é, funciona muito bem para dados utilizados para criá-la, contudo, não é flexível o suficiente para classificar novas amostras.

Para contornar o compromisso existente de viés e variância em qualquer modelagem preditiva, é utilizado método denominado *Ensemble* que envolve grupo de modelos preditivos para conseguir uma melhor acurácia e estabilidade do modelo, através de decisões de múltiplos classificadores para melhorar o desempenho de todo modelo, seus tipos são *Bagging* ou *Bootstrapped*, *Boosting* e *Stacking* (Breiman et al., 1984). Neste estudo, serão focados os modelos *Bagging* e *Boosting*.

#### **3.6.1.3. Florestas Aleatórias (treinamentos múltiplos paralelos)**

Uma implementação de modelo *Bagging* é o algoritmo Florestas Aleatórias, que combina a simplicidade de Árvores de Decisão, com a flexibilidade, resultando em grande

melhoria na acurácia (Breiman, 2001), ou seja, o modelo *Ensemble* combina conjunto de modelos mais simples, como Árvores de Decisão, a fim de obter um resultado generalizado, através do uso de subconjuntos com o propósito de obter uma ideia do conjunto todo (Breiman, 2001). A principal ideia do algoritmo é melhorar a redução da variância do método *Bagging* por diminuição da correlação entre as árvores, sem aumentar muito a variância (Breiman, 2001), isso porque apenas se considera um subconjunto aleatório de variáveis a cada passo (Breiman, 2001).

### 3.6.1.4. Gradient Boosting (treinamentos múltiplos sequenciais)

Conforme descrito acima, Aprendizado de Máquina tem como objetivo criar uma função que se ajuste aos pontos dados ( $x, y$ ), esta função, chamada de modelo, mapeia  $x$  e  $y$ , que pode fazer previsões dados valores de  $x$  desconhecidos (Abu-Mostafa et al., 2012). Adicionar uma série de subfunções para criar uma função composta modeladora de alguns pontos de dados é, então, denominado modelo aditivo.

Árvore Boosting é a combinação de modelos simples, estes são chamados de modelos fracos (Hastie et al., 2001), tipicamente estes modelos são Árvores de Decisão, o algoritmo combina classificadores fracos com intuito de produzir um classificador forte (Hastie et al., 2001). Diferente do *Bagging*, a criação de subconjuntos não é feita de maneira aleatória, e sim feita priorizando subconjuntos mal classificados (Hastie et al., 2001).

Um dos principais algoritmos *Boosting* é o Classificador Gradient Boosting, também conhecido como *Gradient Boosting Machine* (GBM) ou *Gradient Boosted Regression Tree* (GBRT) (Chen & Guestrin, 2016), é um algoritmo que faz a classificação através do modelo aditivo (Friedman, 1999; Hastie et al., 2001). Gradient Boosting faz uso de modelos aditivos para gradualmente aproximar um melhor modelo, de modo a somar submodelos ao modelo composto. Árvores de Decisão tendem a gerar sobreajuste (*overfitting*), e para solucionar este problema, o Gradient Boosting é implementado (Zheng, 2017).

Algoritmo de Gradient Boosting em um conjunto de dados  $\{(x_i, y_i)\}^n$  e uma função de Custo, *Loss Function*  $L(y_i, F(x))$ , estimadora dos parâmetros destas aproximações ajustando os dados, algoritmos de aprendizado usualmente realizam otimização sobre uma função de custo específico, sendo  $y_i$  diferenciável é o valor observado e  $F(x)$  é o valor predito (Friedman, 1999; Hastie et al., 2001).

Primeiro passo iniciar o modelo com valor constante:

$$F_0(x) = \operatorname{argmin} \sum_{i=1}^n L(y_i, \gamma) \quad (\text{Friedman, 1999})$$

$y_i$  é o valor observado e  $\gamma$  são valores preditos.

Essa equação significa que é necessário encontrar valor predito que minimize esta soma. O  $F_0(x)$  representa apenas uma folha da árvore. O passo seguinte é o processo de iteração de todas as  $M$  árvores, número de árvores, iniciando com a primeira árvore  $m = 1$ . Dentro desta iteração, é calculado o valor do resíduo  $r_{im}$ , a diferença entre o valor observado e o predito para cada amostra.

$$r_{im} = \left[ \frac{\delta L(y_i, F(x_i))}{\delta F(x_i)} \right] \text{(Friedman,1999)}$$

São  $m$  amostras para  $i = 1, \dots, n$ ; onde  $n$  é o número de amostras, esta equação representa Gradient Boosting. Posteriormente, dentro do processo iterativo é feito o ajuste de uma árvore de regressão aos valores  $r_{im}$  e criam-se regiões terminais  $R_{jm}$  para cada folha na nova árvore, usando atributos para prever os resíduos  $r$ . A seguir, para cada folha na nova árvore para  $j = 1$  até  $j_m$ , calcula-se:

$$\gamma_{jm} = \operatorname{argmin}_{x_i \in R} L(y_i, F_m(x_i) + \gamma) \text{(Friedman,1999)}$$

valor de saída para cada folha que minimiza a soma, a última etapa do laço iterativo atualiza a função  $F_m(x) = F_{m-1}(x) + v \sum_{i=1}^{J_m} \gamma_{jm} I; x \in R_{jm}$  que é feita nova predição para cada amostra utilizando a predição anterior  $F_{m-1}(x)$ . Nesta etapa, é usado o coeficiente de aprendizado  $v$  redutor do efeito que cada árvore tem na perdição final e tende a melhorar a acurácia a longo prazo. Finalmente, com o fim do ciclo de iteração, retorna-se o valor final de  $F_M$  da última árvore.

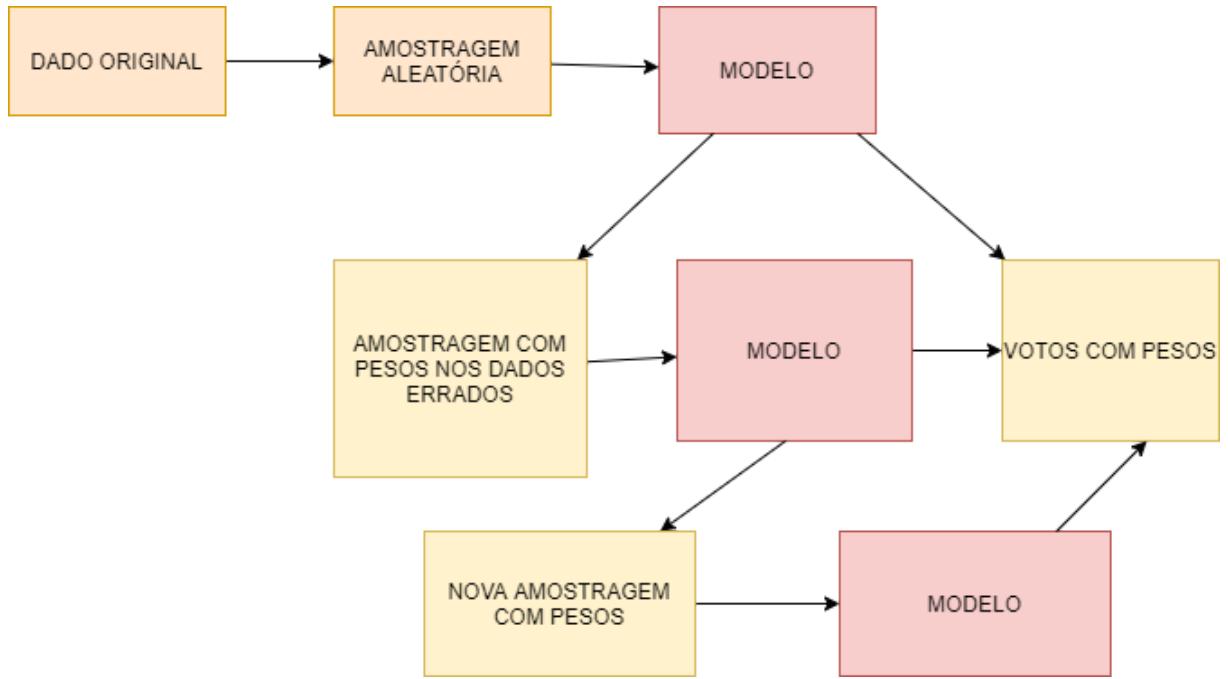


Figura 8: Fluxograma de modelo de aprendizado de Gradient Boosting.

### 3.7. Validação e Avaliação

Em um problema de aprendizado supervisionado, pelo menos dois conjuntos são necessários: o conjunto de treinamento/validação e o conjunto de testes. O modelo é treinado usando dados do conjunto de treinamento resultando em um modelo  $G$ , este modelo, então, prevê as classes para um conjunto de teste através do modelo predito, denominado generalização, com isso estima-se o erro de predição (Hastie et al., 2001).

Objetivo principal não é classificar os dados de treino uma vez que as classes dos dados de treino já são conhecidas, o modelo deve desempenhar bem nos dados desconhecidos ou dados de teste de modo que o intervalo entre o erro de treino e o de teste seja o mínimo possível, tendo dois resultados possíveis: sub-ajuste ou *underfitting* quando o modelo não é capaz de obter um erro de treino baixo suficientemente; e sobreajuste ou *overfitting* ocorre quando o intervalo entre erro de treino e de teste é muito grande; esta heurística é chamada de compromisso viés variância (Hastie et al., 2001). Este compromisso interfere na habilidade de um método de aprendizado generalizar, ou seja, quando a variância é alta implica que o modelo ajusta o ruído aleatório no conjunto de treino, resultando em poder baixo de generalização (*overfitting*) ao passo que modelo com alto viés indica baixa diferença no erro de predição entre o conjunto de treino e conjunto de teste, porém, geralmente possui desempenho ruim (*underfitting*)- Figura 9; (Hastie et al., 2001).

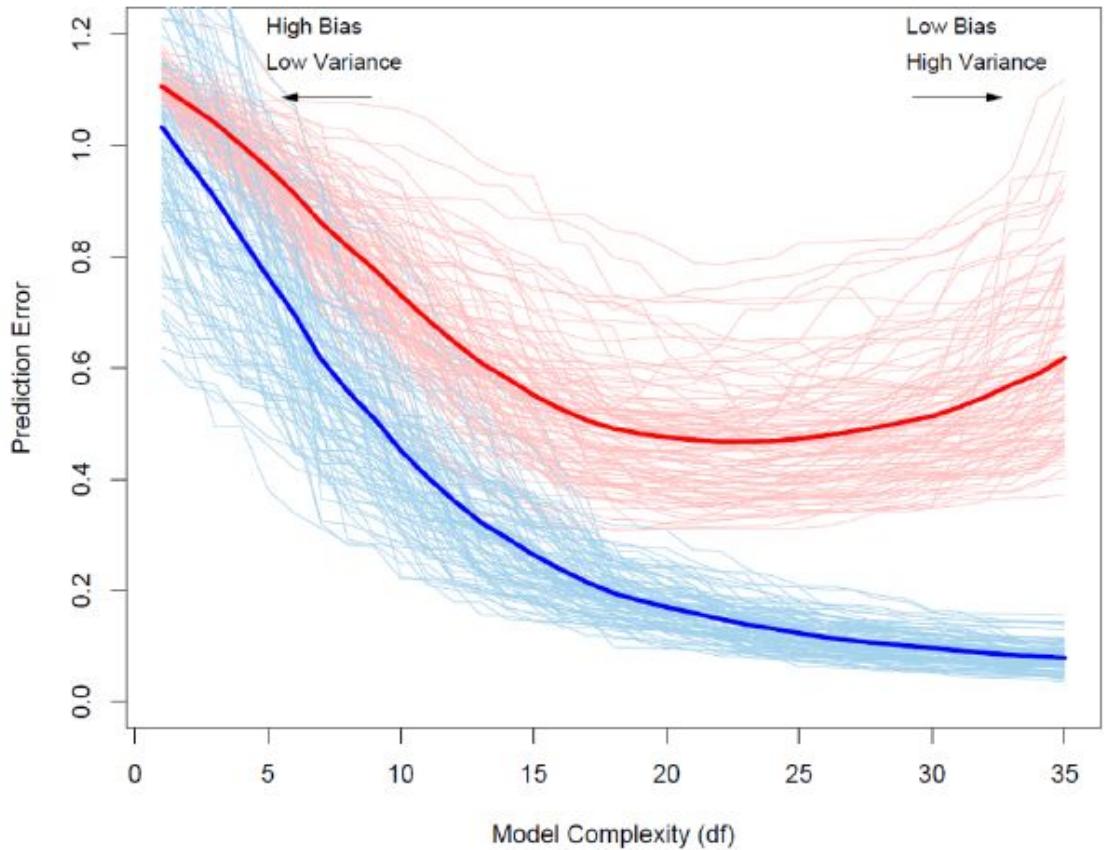


Figura 9: Curvas sólidas de erros de treino (azul) e de teste (vermelho), sendo as médias de todas as curvas. A menor complexidade do modelo com pouca diferença entre as curvas representa sub-ajuste (*underfitting*), e conforme o modelo aumenta sua complexidade, a diferença entre as curvas aumenta e o modelo se encontra em sobre-ajuste (*overfitting*)-neste exemplo as curvas foram calculadas usando 100 diferentes conjuntos de dados com tamanho de 50 cada um; Hastie et al. (2001).

### 3.8. Métrica de classificação

Para quantificar a qualidade do modelo, certas métricas são necessárias, para avaliar os modelos. As principais métricas são acurácia, precisão, recall, F1 (Hossin & Sulaiman, 2015).

Acurácia é um método de métrica de avaliação mais usado para problemas de classificação binária ou de multiclasse, uma vez que a solução produzida é avaliada baseada na porcentagem de previsões corretas sobre o total de instâncias, a métrica complementar de acurácia é a taxa de erro que avalia a solução produzida por sua porcentagem de medições incorretas (Hossin & Sulaiman, 2015).

Precisão e recall são métricas que fornecem mais informações sobre como classificador executa para classes individuais, precisão é a probabilidade de que um dado resultado de classificação para uma amostra realmente pertença a essa classe, ou seja, das classes que foram classificadas como corretas, quantas estão de fato corretas (Hossin & Sulaiman, 2015).

Recall é a probabilidade de uma amostra seja classificada corretamente para uma determinada classe, ou seja, a frequência que o classificador encontra, se de fato pertencer realmente a esta classe. F1 é a média harmônica de precisão e do recall, uma combinação de ambas para dar uma medida única de relevância dos resultados do classificador (Hossin & Sulaiman, 2015).

Matriz de confusão é uma representação matemática destas métricas, usada para descrever o desempenho do modelo, o qual cada entrada da matriz indica a porcentagem de observações de uma dada classe (verdadeira), representada pelas linhas da matriz, identificada como qualquer classe (predita), representada pelas colunas da matriz; e o valor de acurácia pode ser obtido pelos valores da diagonal da matriz de confusão (Bestagini, 2017).

### **3.9. Ajuste do modelo**

Classificação pode ser feita através dos parâmetros padrões e também pode ser obtida através de escolhas de parâmetros ideais ou também chamado de hiperparâmetros, esta última forma é denominada ajuste do modelo ou de parâmetro porque não há fórmula analítica disponível para calcular um valor apropriado para este ajuste (Abu-Mostafa et al., 2012; Hastie et al., 2001).

Os hiperparâmetros controlam a complexidade de um modelo, ajustando o viés e a variância de um algoritmo. Eles não são diretamente estimados dos dados de treinamento, há diferentes métodos para otimizar e buscar hiperparâmetros apropriados para um dado modelo.

A validação para ciência de dados é aplicada através da técnica de amostragem, amplamente usada para estimar o desempenho do modelo, denominada de validação cruzada *K-fold* pelo qual as amostras são aleatoriamente particionadas em um conjunto *k* de aproximadamente mesmo tamanho, e em cada interação com *k-1* destas pastas (*folds*) é usada para treinamento, sendo a pasta restante usada para avaliação do modelo (Abu-Mostafa et al., 2012; Hastie et al., 2001). Outro método é o *leave-one-out*, um caso específico do *k-fold*, com *k* igual ao número total de dados. Nesta abordagem, é realizado um número de cálculos de erro proporcional ao número dados, um para cada dado. Apesar de apresentar uma investigação completa sobre a variação do modelo em relação aos dados utilizados, este método possui um alto custo computacional, mais indicado para situações nas quais poucos dados estão disponíveis (Abu-Mostafa et al., 2012; Hastie et al., 2001). Existem outros métodos de validação que este projeto não abordou como *leave p out* e *holdout*.

Além disso, outro parâmetro importante é o fator de regularização que define quanto se pode evitar classificações incorretas, de modo que um razoável fator de regularização tentará classificar corretamente mais exemplos dos dados de treinamento. No entanto, um fator de regularização muito elevado pode ocasionar *overfitting* dos dados e falhar em generalizar para dados de teste, ao passo que um fator de regularização muito pequeno não será satisfatório no ajuste de *outliers* e terá um grande erro no conjunto de treinamento (Abu-Mostafa et al., 2012; Hastie et al., 2001).

De forma sucinta, regularização é uma forma de evitar o *overfitting*, buscando penalizar parâmetros muito grandes para que seus crescimentos não prejudiquem a capacidade de generalização do modelo (Abu-Mostafa et al., 2012; Hastie et al., 2001).

Portanto, o ajuste de parâmetros é uma parte crítica do fluxo de trabalho quando se busca construir um modelo de Aprendizado de Máquina.

#### **4. RESULTADOS OBTIDOS**

Os resultados gerados foram produzidos através da implementação de códigos de programação em linguagem Python e suas bibliotecas como Pandas, Numpy e Matplotlib (Hall, 2016). Foi feita a análise exploratória dos dados, verificando a distribuição estatística dos dados, visualização dos perfis dos poços e gráficos de correlação entre as variáveis do estudo. Também, foi feito o pré-processamento dos dados com tratamento de *features*, com aplicação da técnica de PCA e FE. Ademais, o algoritmo escolhido foi Gradient Boosting que desempenhou muito bem em comparação a outros métodos como Máquina de Vetores Suporte, Redes neurais, Florestas Aleatórias, Adaboost (Hall, 2016). A partir da validação cruzada, divisão do número de dados de treinamento/validação e teste, encontram-se parâmetros ótimos para o modelo, processo denominado *grid searching*, fazendo com que o modelo seja treinado, para atingir uma avaliação final para novos dados de teste (Hastie et al., 2001).

Tendo como base os trabalhos implementados por Hall (2016), os resultados obtidos foram adquiridos importando as bibliotecas de Python como Numpy, Pandas e Matplotlib, Seaborn e Sklearn. Com Pandas, versão 1.1.5, é possível ler os dados que estão no formato de uma planilha excel.csv através da criação de um Data Frame (VanderPlas, 2016) na forma de uma tabela de 4149 linhas e 11 colunas, Tabela 2.

Tabela 2: Conjunto de dados de treinamento pela biblioteca Pandas (Hall, 2016).

	Facies	Formation	Well Name	Depth	GR	ILD_log10	DeltaPHI	PHIND	PE	NM_M	RELPOS
0	3	A1 SH	SHRIMPLIN	2793.0	77.450	0.664	9.900	11.915	4.600	1	1.000
1	3	A1 SH	SHRIMPLIN	2793.5	78.260	0.661	14.200	12.565	4.100	1	0.979
2	3	A1 SH	SHRIMPLIN	2794.0	79.050	0.658	14.800	13.050	3.600	1	0.957
3	3	A1 SH	SHRIMPLIN	2794.5	86.100	0.655	13.900	13.115	3.500	1	0.936
4	3	A1 SH	SHRIMPLIN	2795.0	74.580	0.647	13.500	13.300	3.400	1	0.915
...	...	...	...	...	...	...	...	...	...	...	...
4144	5	C LM	CHURCHMAN BIBLE	3120.5	46.719	0.947	1.828	7.254	3.617	2	0.685
4145	5	C LM	CHURCHMAN BIBLE	3121.0	44.563	0.953	2.241	8.013	3.344	2	0.677
4146	5	C LM	CHURCHMAN BIBLE	3121.5	49.719	0.964	2.925	8.013	3.190	2	0.669
4147	5	C LM	CHURCHMAN BIBLE	3122.0	51.469	0.965	3.083	7.708	3.152	2	0.661
4148	5	C LM	CHURCHMAN BIBLE	3122.5	50.031	0.970	2.609	6.668	3.295	2	0.653

4149 rows × 11 columns

Além disso, outra tabela foi gerada, Tabela 3, que descreve a distribuição estatística de *features* de entrada, mostrando número de vetores de *features*, média, desvio padrão, mediana, quartil, percentil, valores máximos e mínimos do conjunto de dados de treinamento; e o tratamento estatístico dos dados (Figuras 11,12 e 13).

Tabela 3: Distribuição estatística das variáveis/features por Pandas (Hall, 2016).

	Facies	Depth	GR	ILD_log10	DeltaPHI	PHIND	PE	NM_M	RELPOS
count	4149.000000	4149.000000	4149.000000	4149.000000	4149.000000	4149.000000	3232.000000	4149.000000	4149.000000
mean	4.503254	2906.867438	64.933985	0.659566	4.402484	13.201066	3.725014	1.518438	0.521852
std	2.474324	133.300164	30.302530	0.252703	5.274947	7.132846	0.896152	0.499720	0.286644
min	1.000000	2573.500000	10.149000	-0.025949	-21.832000	0.550000	0.200000	1.000000	0.000000
25%	2.000000	2821.500000	44.730000	0.498000	1.600000	8.500000	3.100000	1.000000	0.277000
50%	4.000000	2932.500000	64.990000	0.639000	4.300000	12.020000	3.551500	2.000000	0.528000
75%	6.000000	3007.000000	79.438000	0.822000	7.500000	16.050000	4.300000	2.000000	0.769000
max	9.000000	3138.000000	361.150000	1.800000	19.312000	84.400000	8.094000	2.000000	1.000000

As variáveis são processadas e é realizada a atribuição de classes das fácies definidas por cores para cada tipo de rocha de modo que os *logs* são ordenados pela profundidade de forma crescente por Hall (2016)- Figura 10.

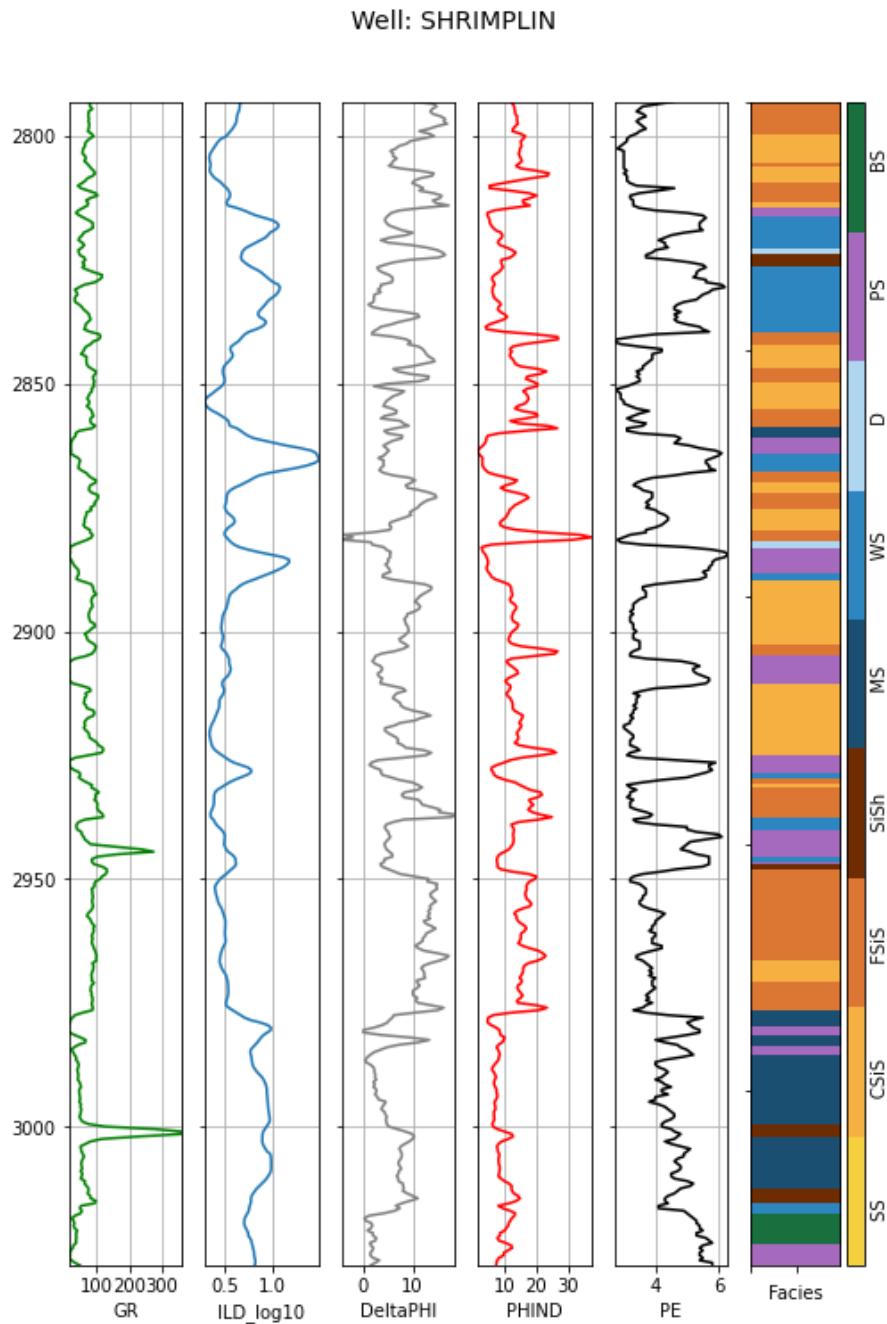


Figura 10: Perfil de poço SHRIMPLIN e as cinco variáveis: raios gama, resistividade, porosidade, densidade e efeito fotoelétrico; à direita interpretação do poço com oito fácies representadas em cores diferentes (Hall, 2016).

É importante visualizar como as fácies estão distribuídas numericamente através do conjunto dos dados de forma completa em um histograma, Figura 11, a observar um maior número de exemplos, 649, de rochas de siltito grosso não marinho (CSiS) e um menor número de rochas de dolomito (D), 81 exemplos.

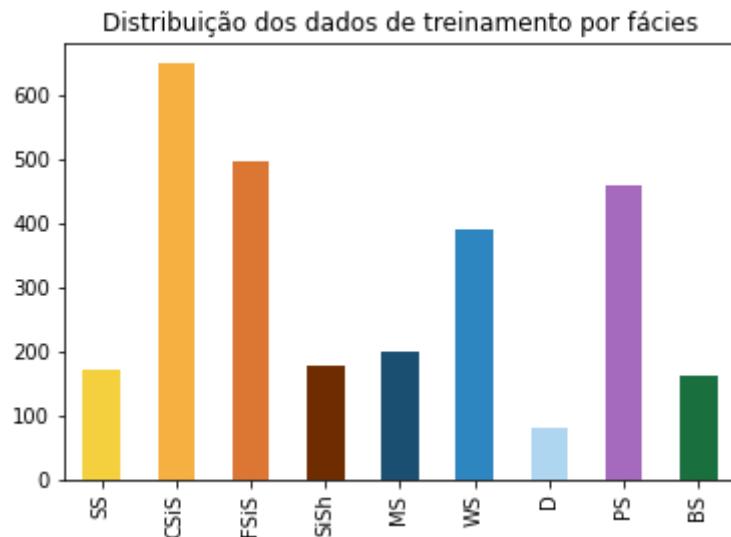


Figura 11: Histograma dos dados de treinamento desbalanceados que mostra o número de exemplos em cada fácie, destaque na diferença entre fácies 2 e 7 (Hall, 2016).

Pela distribuição de fácies de cada poço (Figura 12), verifica-se que os exemplos de fácies 9, bafflestone filóide-algal (BS) são muito baixos. É preciso comentar que o poço Recruit F9 (Hall, 2016) foi criado artificialmente, pseudo poço, para representar este tipo de rocha. Ademais, somente alguns poços contêm amostras para todas as classes.

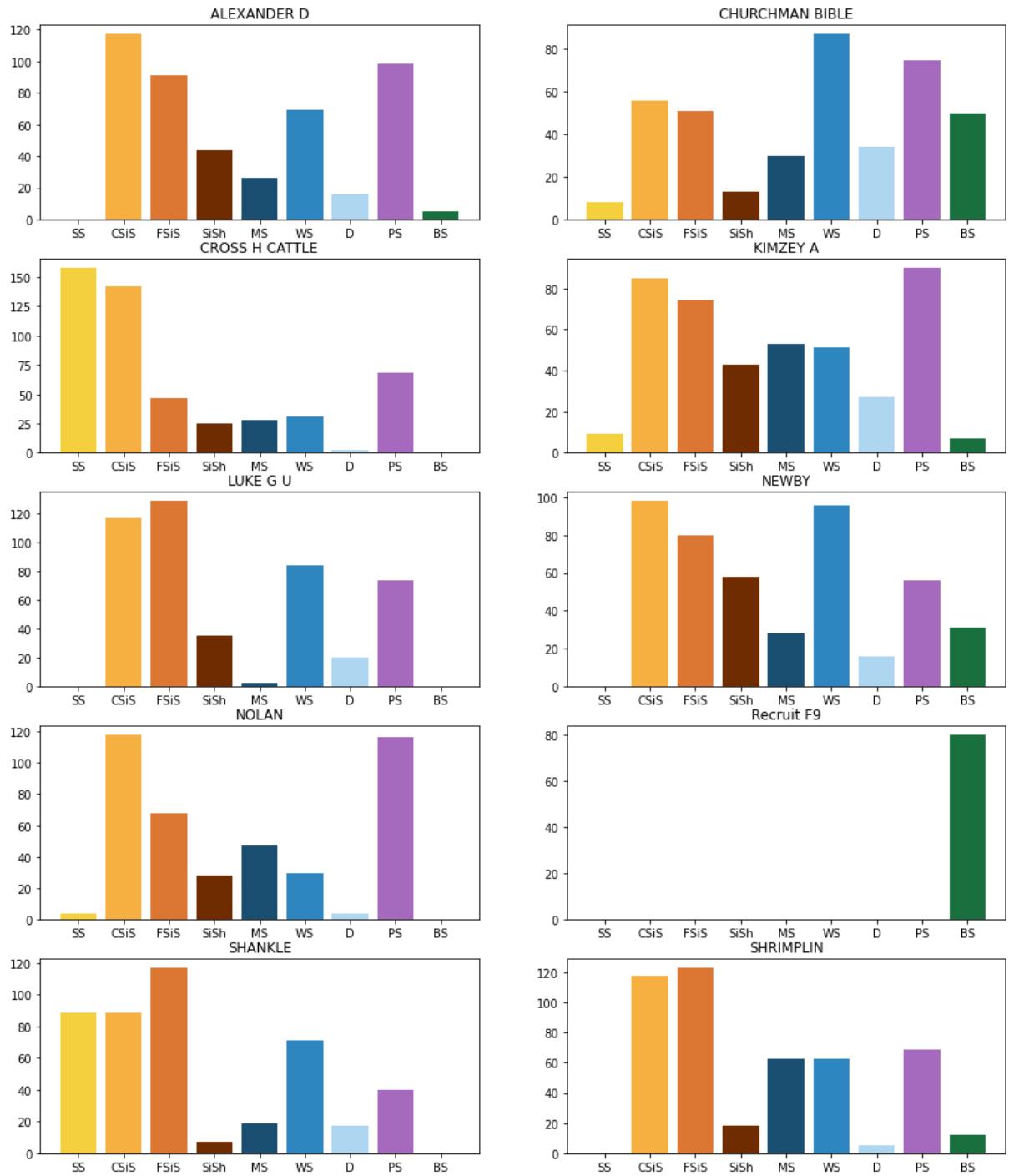


Figura 12: Distribuição de fácies por poços (Bestagini et al., 2017).

Além disso, pela distribuição de *features* de cada poço (Figura 13), observa-se ausência da variável efeito fotoelétrico (PE) nos poços Kimzey A, Alexander D e Recruit F9.

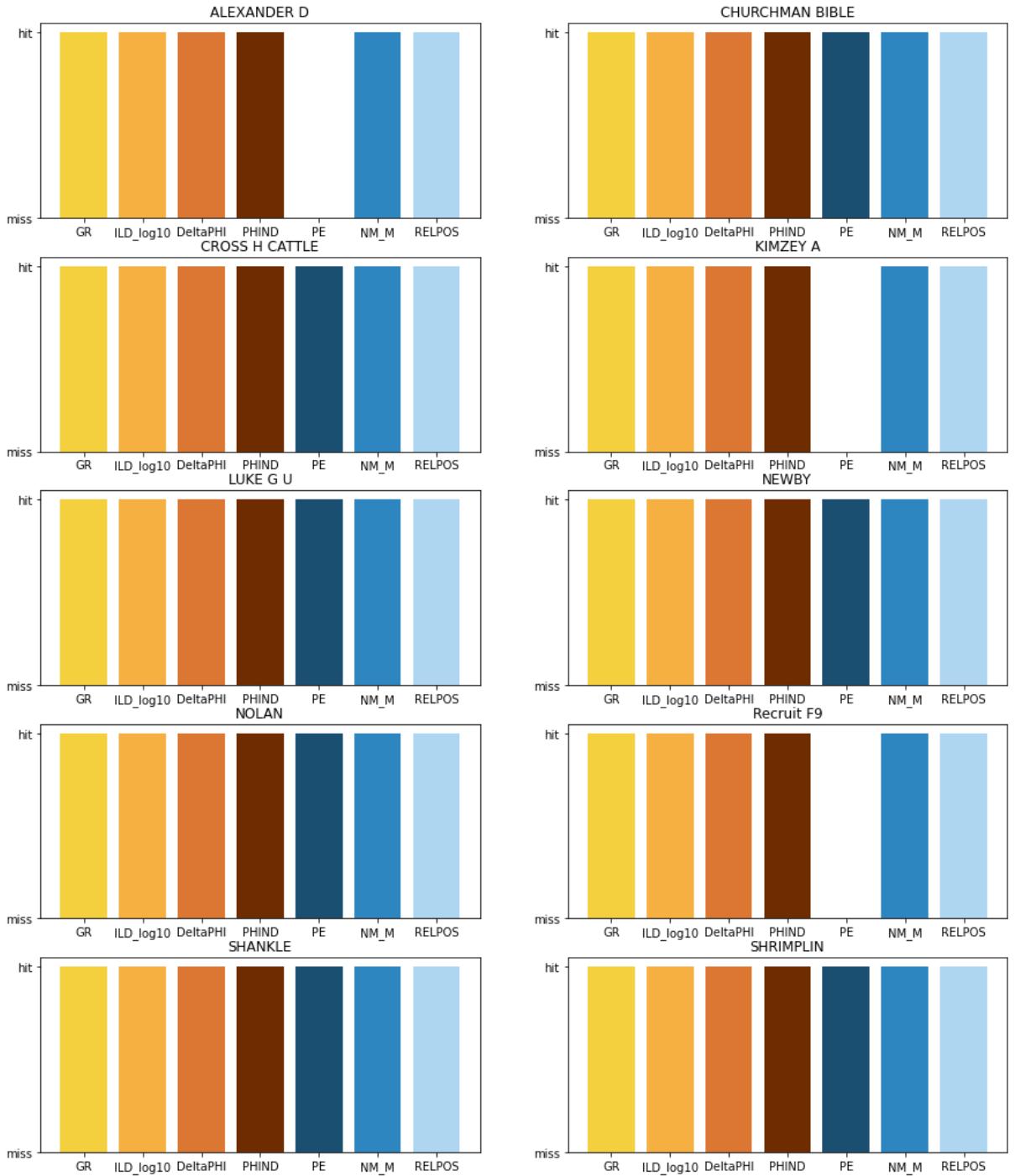


Figura 13: Distribuição de features dos poços do estudo, mostrando se algum poço possui ou não determinada característica (Bestagini et al., 2017).

A seguir, a Figura 14 mostra gráficos 'crossplots', uma ferramenta, pela biblioteca de gráficos Seaborn do Python, muito utilizada em geociências para visualizar como duas propriedades variam com determinada rocha. A denominada matriz de dispersão pode ajudar a visualizar rapidamente a correlação entre as 5 variáveis-raios gama (GR), registro de resistividade (ILD\_log10), efeito fotoelétrico (PE), diferença de porosidade de densidade de nêutrons e porosidade de densidade de nêutrons média (DeltaPHI e PHIND); foram

excluídas neste gráfico as indicadoras não marinho-marinho (NM\_M) e posição relativa (RELPOS). Cada painel do gráfico, abaixo e acima da diagonal principal, mostra a relação entre duas das variáveis nos eixos x e y, com cada ponto colorido de acordo com suas fácies. O mesmo mapa de cores é usado para representar as 9 fácies.

A diagonal principal contém a curva distribuição de densidade de probabilidade de cada *feature*, notam-se a sobreposição de classes de *features*, sendo difícil distinguir qual variável possui determinada classe, ou seja, é difícil assumir com grande certeza que determinado ponto pertença de fato à cor amarela, fácie 1, por exemplo.

A correlação entre estas variáveis mostra uma evidente sobreposição e difícil separação das classes, principalmente, entre as fácies 2, de maior quantidade, e fácie 3, isso se deve, provavelmente, à diferença de granulometria entre o siltito grosso e o fino.

Nota-se que alguns *features* parecem ser afetados por *outliers*, ou seja, fácies que estão nas extremidades, distantes de agrupamentos de pontos.

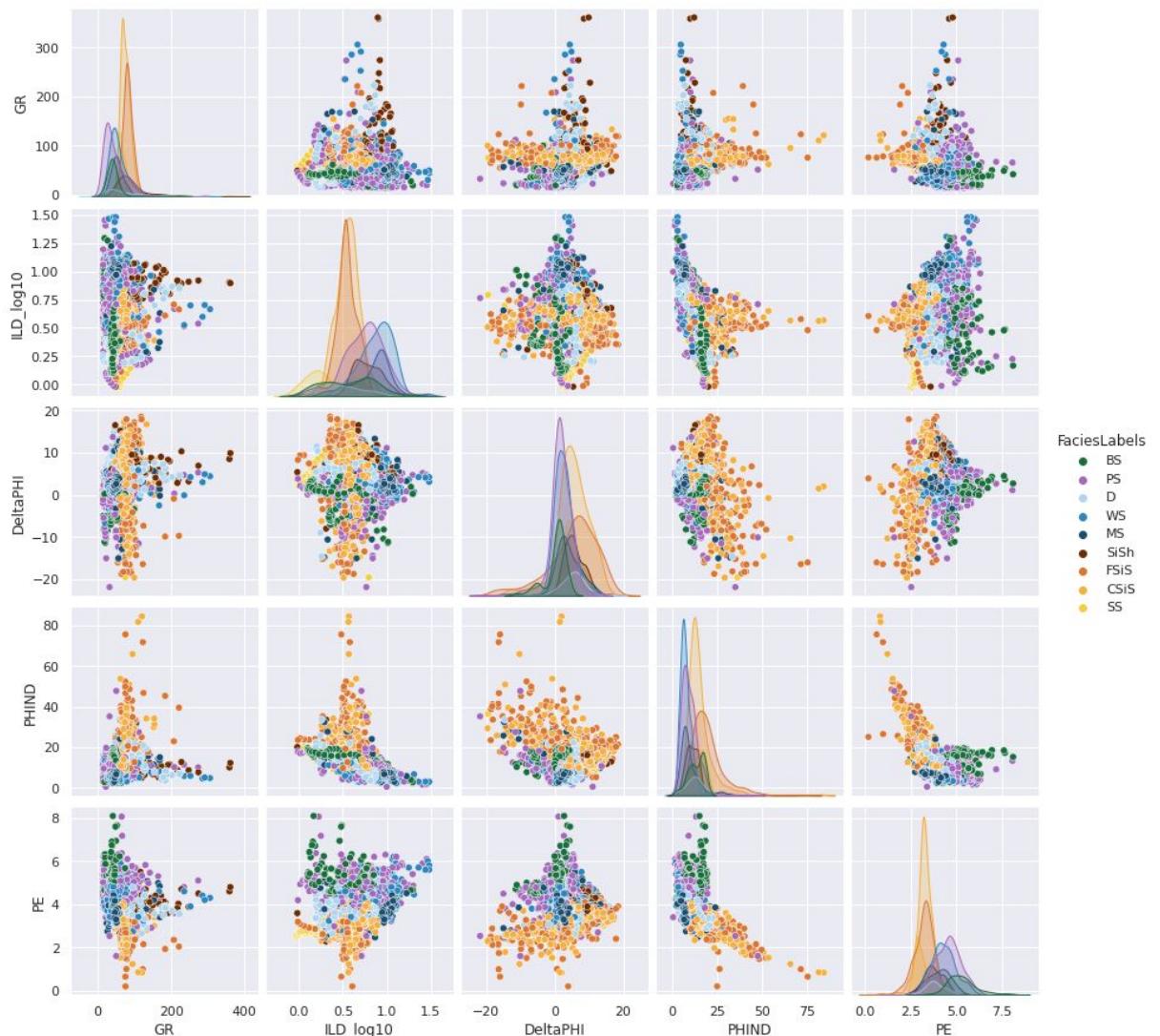


Figura 14: Gráficos crossplots de classes e features gerados por Seaborn (Hall, 2016).

É feita aplicação de PCA, Análise de Principais Componentes, para visualizar melhor as classes, dividindo-as quanto a litologia em rochas clásticas (fácies 1,2,3,4,5) e carbonáticas (fácies 6,7,8,9)- Figura 15; tipos marinhos (fácies 1,2,3) e não marinhos (4,5,6,7,8,9)- Figura 16; ambientes deposicionais clástico não marinho (1,2,3), clástico marinho (4), plataforma carbonática (5,6,7,8,9)- Figura 17; tendo como base o modelo geológico e estratigráfico da região. Essas classes definidas em novos eixos PCA1 e PCA2 ( $X_{pca\_1}$  e  $X_{pca\_2}$ ) são a representação matemática de todo treinamento de *features* após a aplicação de PCA. A técnica de PCA foi implementada através da biblioteca Sklearn bem como outras bibliotecas Python para manipulação do processo de FE.

Novamente, uma implicação disso é o uso mais refinado de técnicas de FE.

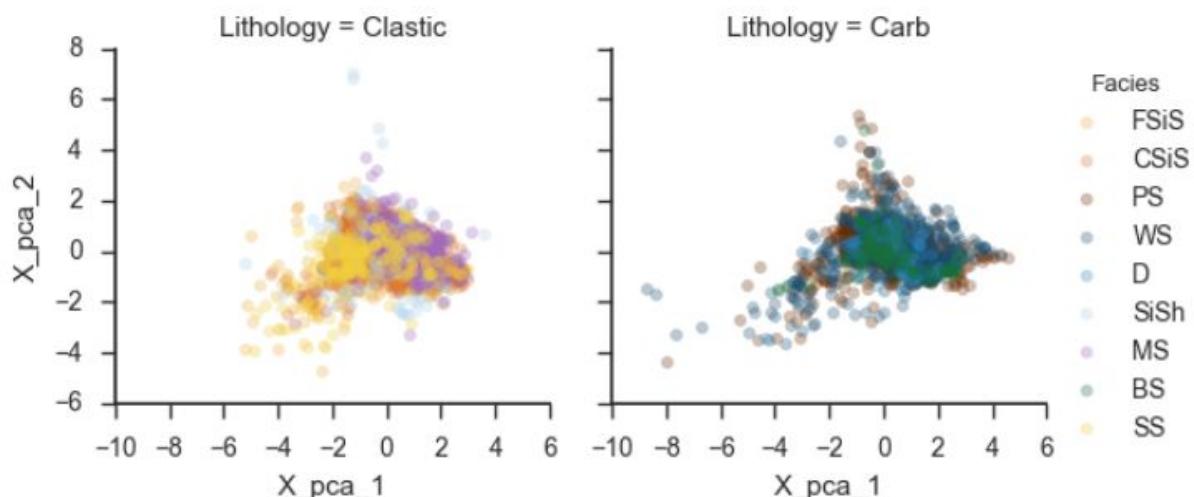


Figura 15: Separação em duas classes por litologia por aplicação de principais componentes (PCA).

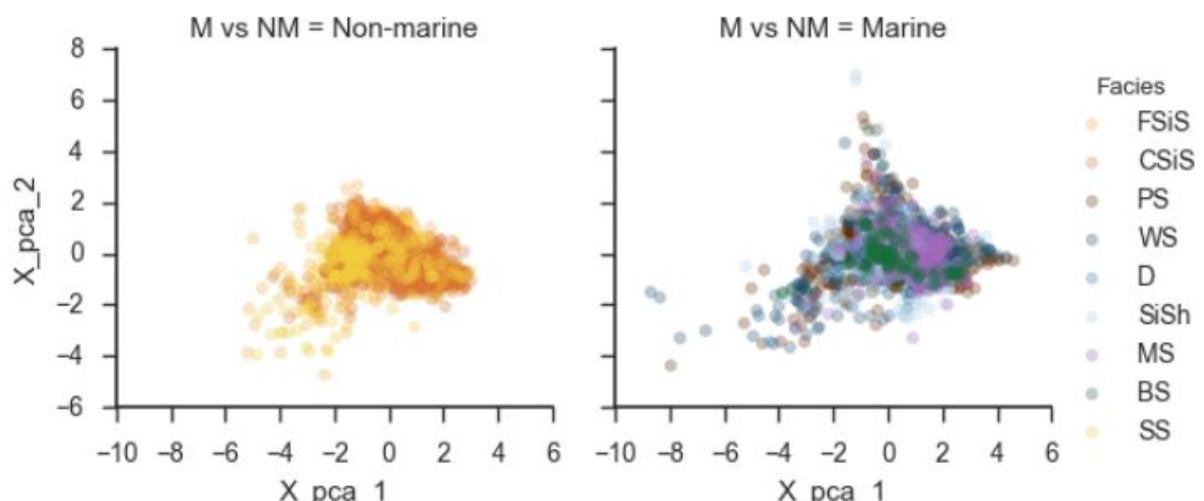


Figura 16: Tipos marinhos e não marinhos de rocha existentes no problema.

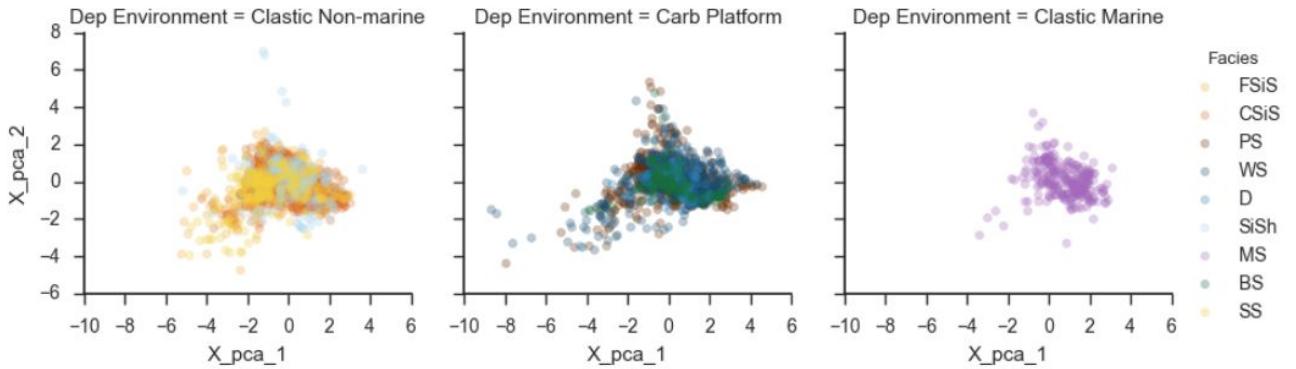


Figura 17: Separação das classes por ambiente deposicional (PCA).

Como foi observada na análise exploratória dos dados, não há valores de *feature* PE, efeito fotoelétrico, em três dos dez poços. Uma forma de completar estes dados é pela média dos valores de PE (Bestagini et al., 2017) ou usando dados de treinamento de valores de sete bons poços e predizer valores dos outros dois poços que faltam por regressão linear (Guardo, 2018). Dessa forma, optou-se neste estudo pela média e pelo uso de todos os 10 poços.

O algoritmo Gradient Boosting considerou uma floresta de 100 árvores, cada uma alimentada com não mais do que 10 *features*, e adotada a estratégia um contra um para lidar com problemas de multi classes , no caso, de 9 fácies (Bestagini, 2017).

A validação foi feita pelo método validação cruzada 10-fold, em que cada poço é um fold, ou seja, são selecionados os dados do conjunto de treinamento de 9 poços, em média 3734 exemplos, para treinar o classificador, então, testa-se no décimo poço, em média 415 dados, de modo que o procedimento de teste e treino é repetido dez vezes, misturando os poços a fim de construir diferentes conjuntos de treino e teste com cada poço de uma vez apenas, os resultados são calculados em média ao longo de 10 experimentos (Bestagini, 2017).

Definem-se os parâmetros de otimização tais como: seleção da divisão dos dados, normalização de *features*, para ter variáveis preditoras no mesmo intervalo para poder comparar diversos perfis, usando um dimensionador robusto, principalmente para *outliers*, treinamento do classificador, teste do classificador treinado nos dados de validação, repetido para todas as divisões e obtendo a média dos F1 score, com a expectativa do classificador ser capaz de generalizar bem o conjunto de teste. A partir deste método, geram-se duas matrizes de confusão (Figuras 18 e 19), que serão explicadas na próxima seção.

Se for necessário verificar pontualmente o desempenho de um poço nos dados de treino, assume-se um poço não visualizado em relação aos demais poços, o *blind well*, este poço serve de comparação para avaliar se a sequência produzida no treinamento está semelhante ao poço verdadeiro (Guardo, 2018) (Figuras 20 e 21), detalhado na próxima seção.

Aplica-se, após este processo, o modelo de classificação para dados de teste sem as classes, para verificar se houve significativa generalização dos dados no modelo, uma vez treinado o modelo de classificação, pode-se usar o modelo para identificar poços que não têm dados de testemunho, ou ainda, para qualquer número de poços que tenha mesmo conjunto de perfis de poços, variáveis preditoras, como entrada, fazendo, então, predição de classes de dados sem classes. No caso deste estudo, serão aplicados para dois poços STUART e CRAWFORD- Figuras 22 e 23.

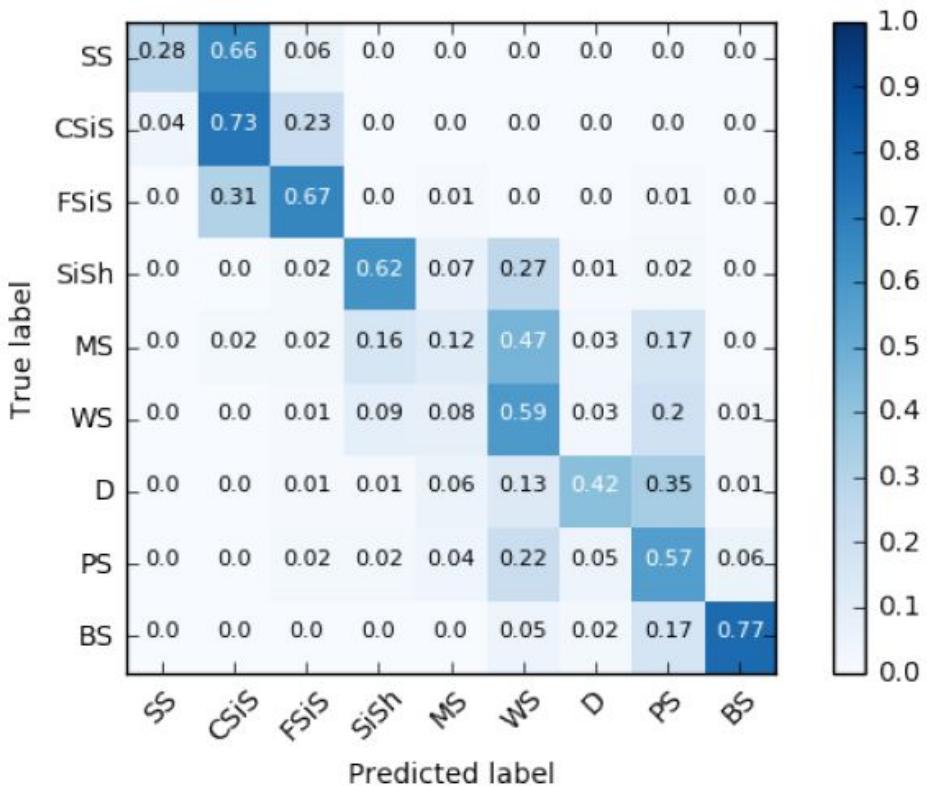


Figura 18: Matriz de confusão produzida da validação com *feature augmentation*, destaque na diagonal para CSiS e BS -73% e 77% das amostras foram preditas como verdadeira, entretanto, calcários MS com 12% e WS se confundem na predição (Bestagini et al., 2017).

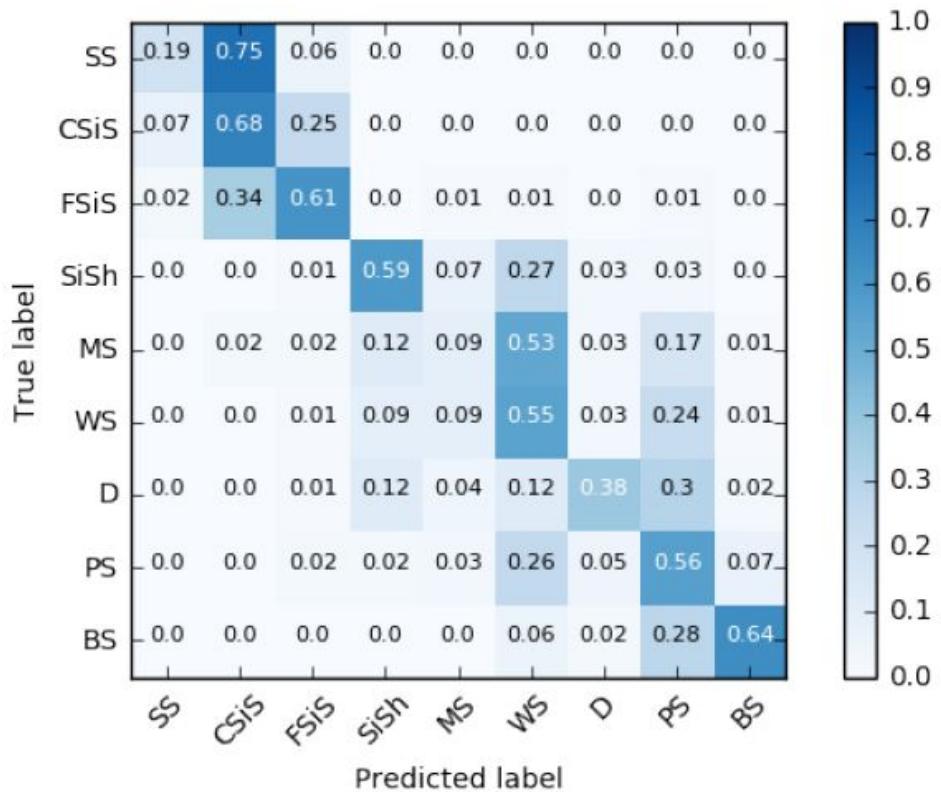


Figura 19: Matriz de confusão sem *feature augmentation*, note que as porcentagens de predição diminuíram em relação a outra matriz (Bestagini et al., 2017).

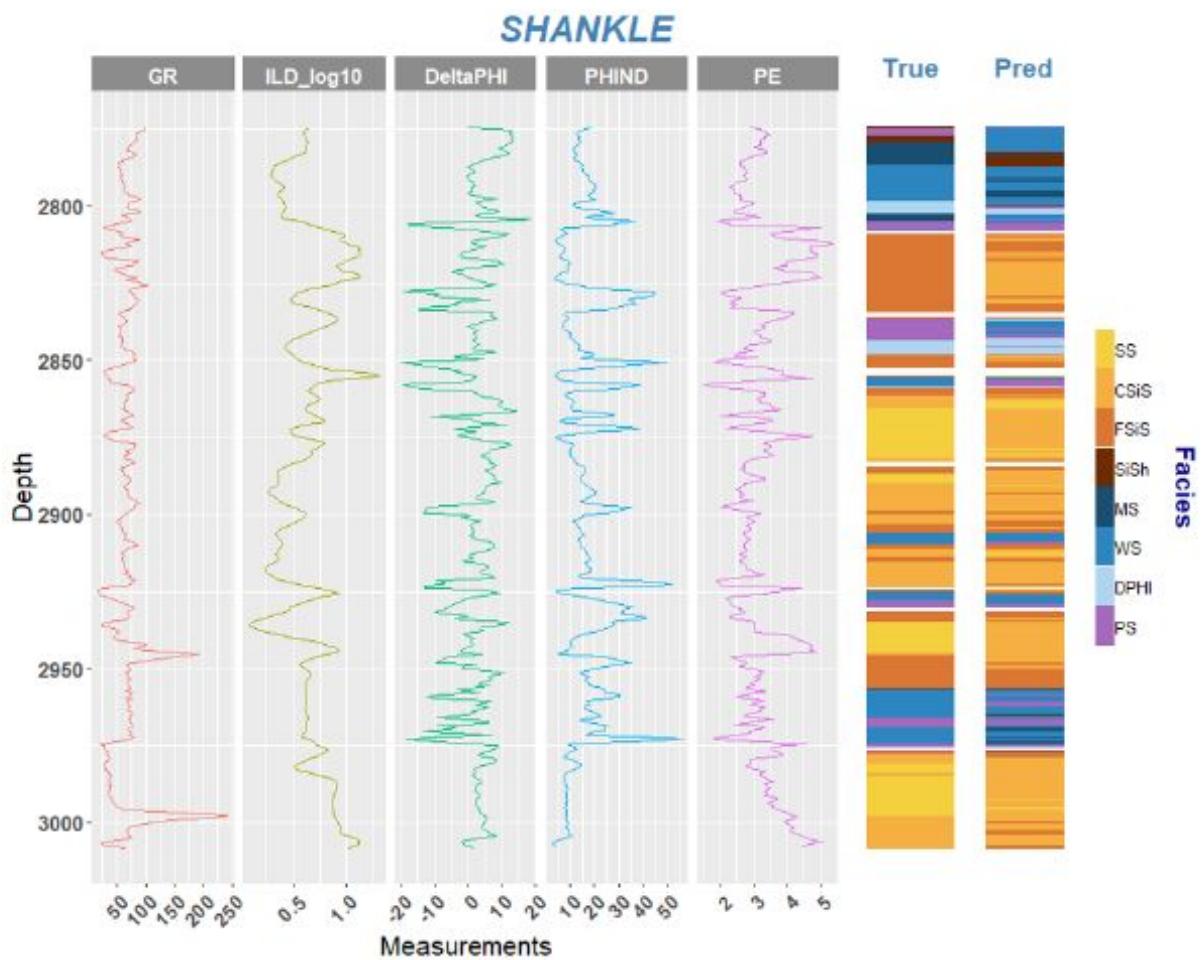


Figura 20: Poço Shankle colocado como blind well, para verificar o modelo preditivo no conjunto de dados de treinamento comparado a fácies verdadeiras, resultado gerado sem realizar o pré-processamento das variáveis preditoras (Guarido, 2018).

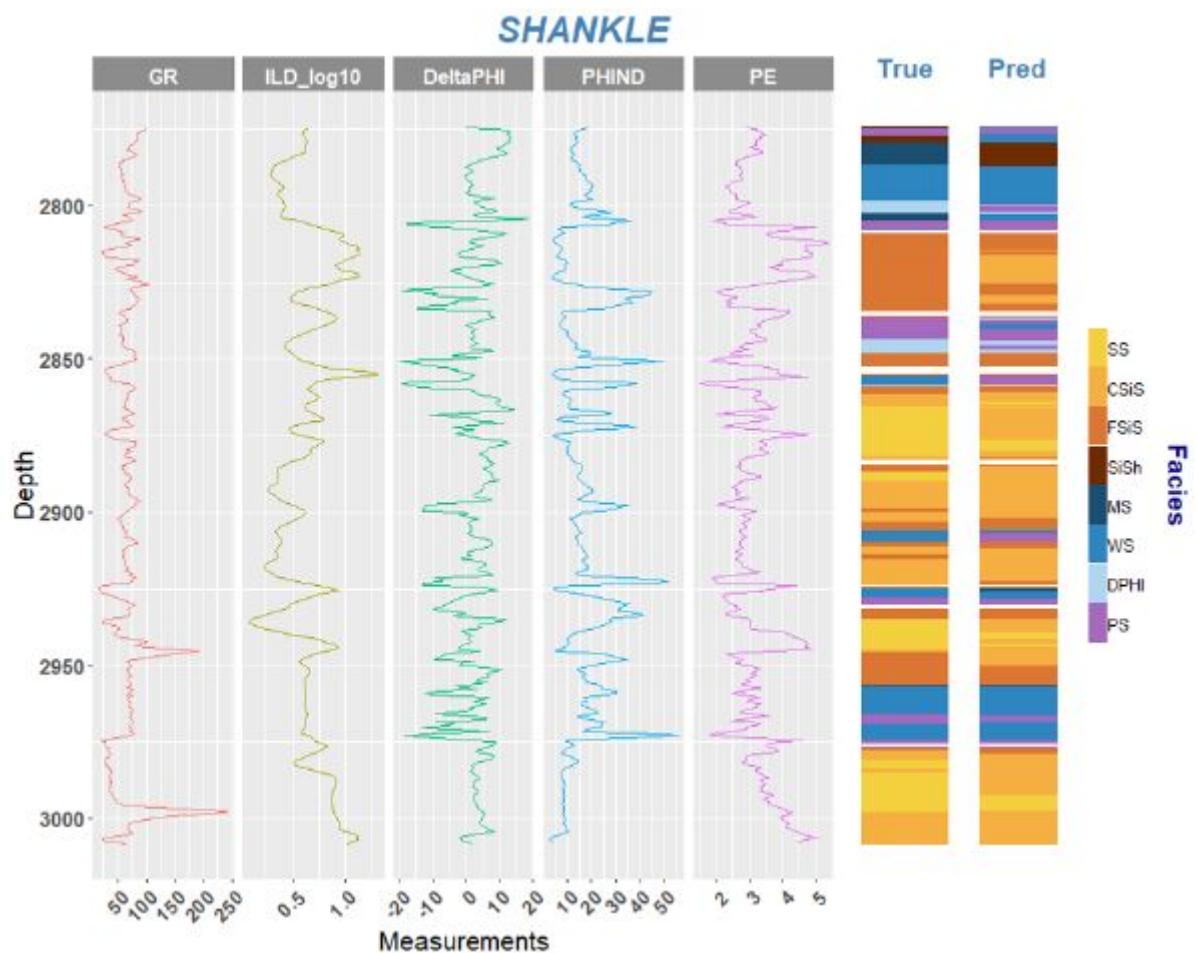


Figura 21: Poço Shankle colocado como blind well, para verificar o modelo preditivo com ajustes de parâmetros de otimização no conjunto de dados de treinamento comparado a fácie verdadeiras, resultado gerado com o pré-processamento das variáveis preditoras (Guardado, 2018).

Well: STUART

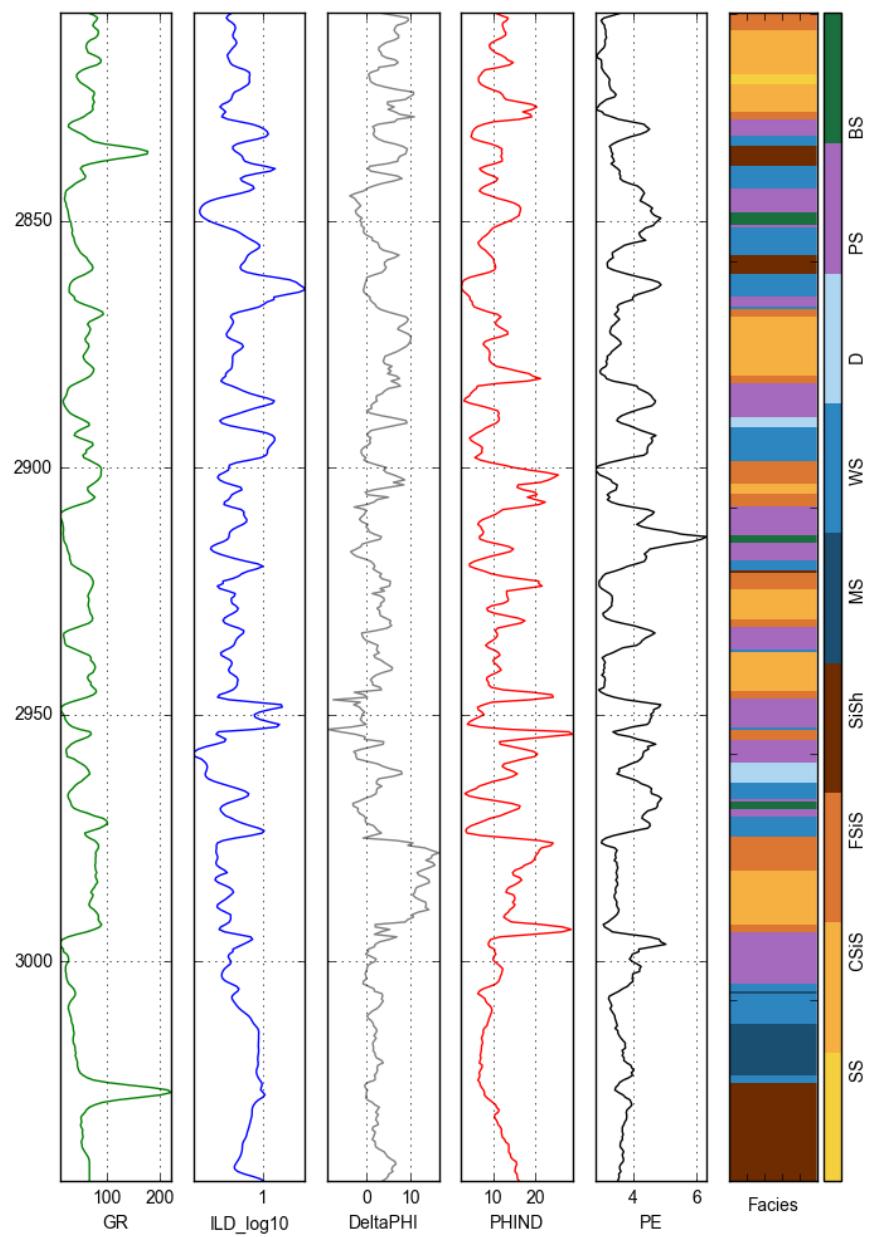


Figura 22: Poço STUART gerado da aplicação do modelo de classificação para dados de teste sem classes (Bestagini,2017).

### Well: CRAWFORD

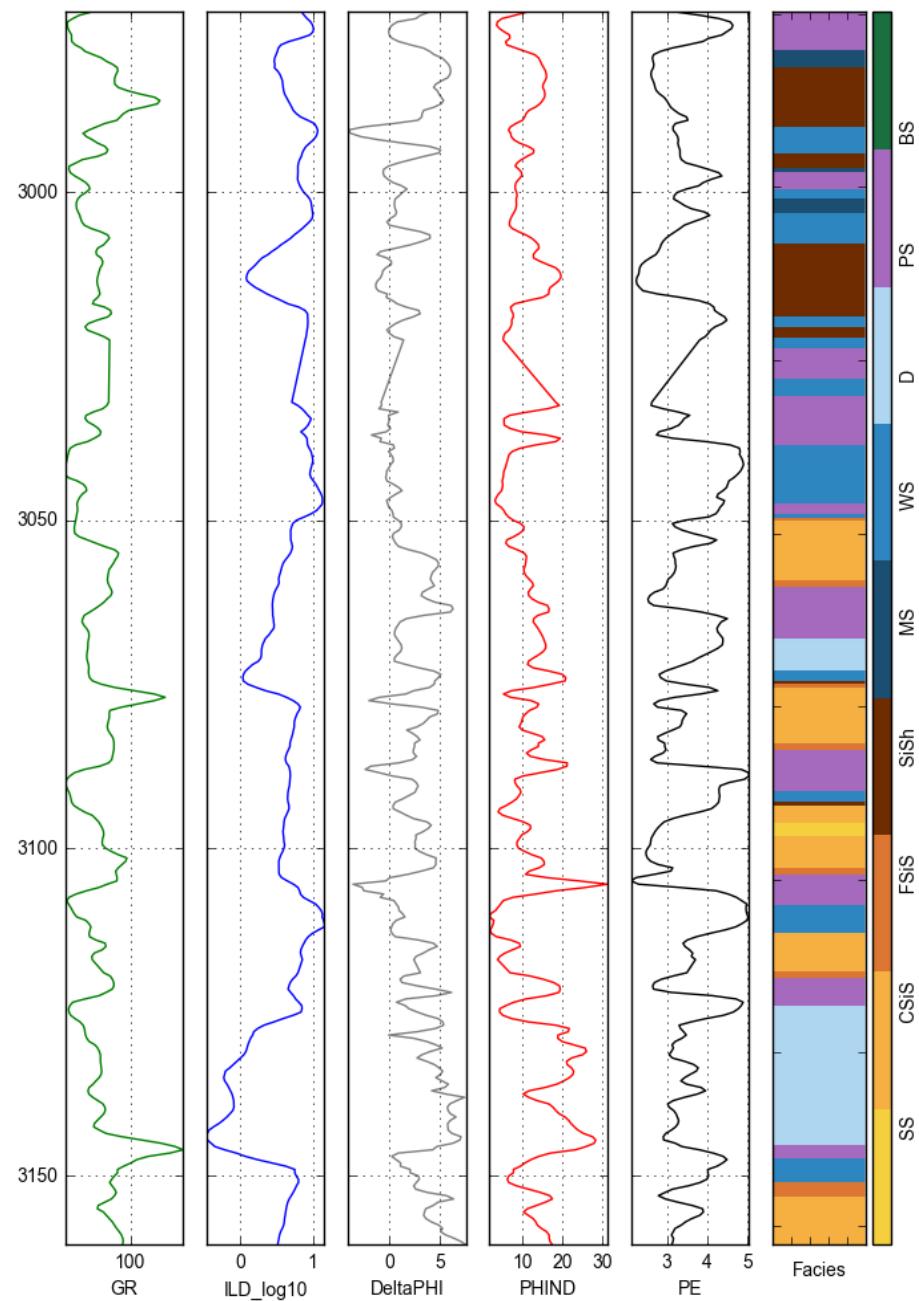


Figura 23: Poço CRAWFORD gerado da aplicação do modelo de classificação para dados de teste sem classes (Bestagini,2017).

## 5. INTERPRETAÇÃO E DISCUSSÃO DOS RESULTADOS

A comparação feita entre as sequências de fácies preditas e verdadeiras no poço Shankle como *blind well* mostra a diferença entre o não uso de dados pré-processados (Figura 20) e quando aplicam-se métodos de tratamento de *features* e ajustes de parâmetros de otimização (Figura 21) (Guardo, 2018). Ainda que na parte rasa do perfil em ambas as maneiras, há inconsistências na classificação. Portanto, modelos ajustados e com melhor tratamento nos dados possuem maior consistência na classificação (Guardo, 2018), entretanto, uma análise mais completa do desempenho é feita através da matriz de confusão.

O resultado produzido da Figura 18 mostra que algumas fácies possuem de fato mais facilidade para classificar do que outras, por exemplo, é possível corretamente identificar que 77% de bafflestone (BS) e 73% de siltito grosso não marinho (CSiS), isto é, estas porcentagens na diagonal da matriz representam quanto foi previsto para cada fácie de maneira correta. Por outro lado, mudstone (MS) é mais identificado como wackstone (WS), grupo de calcários, ou seja, 47% de WS, predito nesta classe, são na verdade MS. Fácies marinhas (SiSh, MS, WS, D, PS, BS) são classificadas de forma incorreta em relação a outras fácies marinhas, da mesma forma que fácies não marinhas (SS, CSiS, FSiS), em relação a outras não marinhas (Bestagini, 2017). Um exemplo disso é o que se refere à granulometria, para siltitos não marinhos grosso e fino - CSiS e FSiS, o algoritmo não consegue diferenciar de forma eficiente, produzindo grande erro no modelo, mesmo a análise petrográfica de testemunho é um desafio para interpretação. A Figura 14 da matriz de dispersão reitera a dificuldade de interpretação de *features* dos perfis de poços, visto que estão bastante superpostos. E raramente, fácies marinhas são confundidas com não marinhas e vice-versa.

Ademais, a matriz de confusão sem o método de *feature augmentation* foi mostrada a título de comparação (Figura 19), observa-se que a diagonal principal possui valores menores em relação a matriz de confusão com a técnica aplicada. Em termos de f-score, o algoritmo alcançou 0,61 em média sobre o disponível 10 *folds*, este valor cai para 0,55 sem a aplicação de *feature augmentation* (Bestagini, 2017). Uma maneira de aumentar a acurácia é considerar as fácies adjacentes, o que aumenta para cerca de 0,93 (Hall, 2016), uma vez que se considera que as fácies não mudam abruptamente de uma dada profundidade para uma próxima (Hall, 2016), a mesma ideia para realizar *feature augmentation* quando se considera *features* de camadas vizinhas estarem de alguma forma correlacionadas.

Uma explicação porque algumas fácies são melhores classificadas do que outras está na Figura 11 que mostra número de fácies por poços, em particular MS que possui um número de amostras menor, fazendo com que o algoritmo não aprenda significamente como caracterizá-las, ao passo que o maior número de exemplos de CSiS explica o maior desempenho de classificação, pois as classes estão desbalanceadas. Dessa forma, é necessário ter um maior conjunto de poços para que o algoritmo possa alcançar um desempenho melhor no aprendizado e também teria sido importante ter aplicado uma técnica de balanceamento entre as classes para evitar essa diferença de desempenho, isto poderá ser feito em trabalhos futuros. Outra forma de reduzir problemas de sobreposição de *features* e em relação o contato entre as fácies é selecionar as porções das classes mais representativas que também pode ser desenvolvido futuramente.

Na parte de generalização do modelo dos dados de teste para poços sem classes, as figuras 24 e 25 mostram a comparação entre os poços gerados do modelo e os poços verdadeiros.

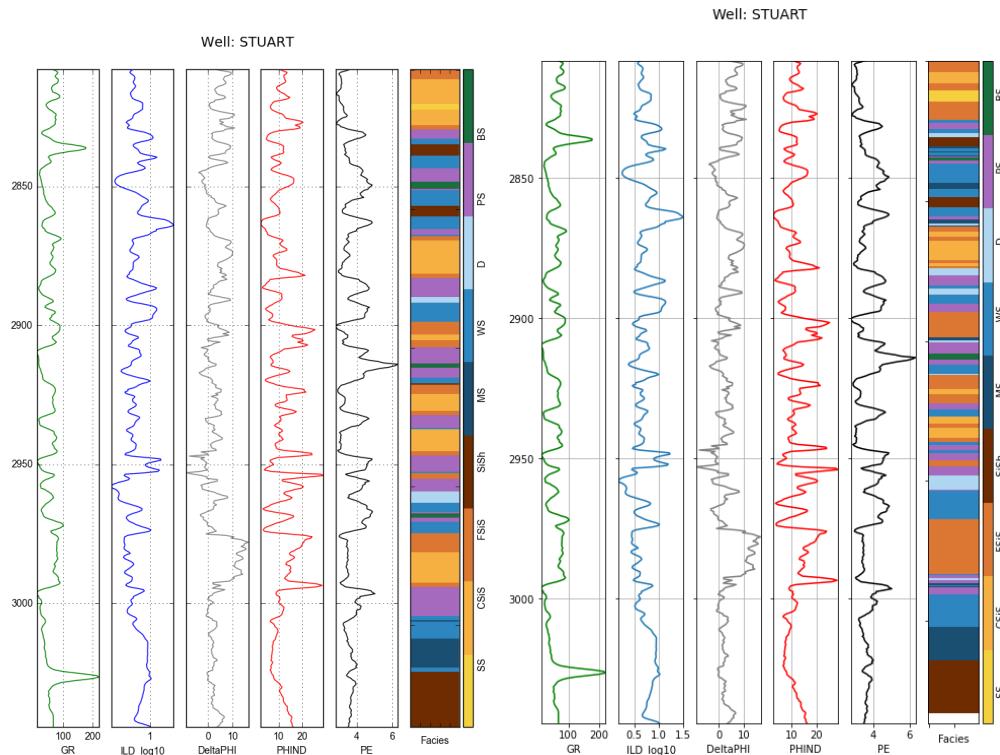


Figura 24: Poço STUART gerado por predição à esquerda e o poço verdadeiro à direita (Bestagini et al., 2017).

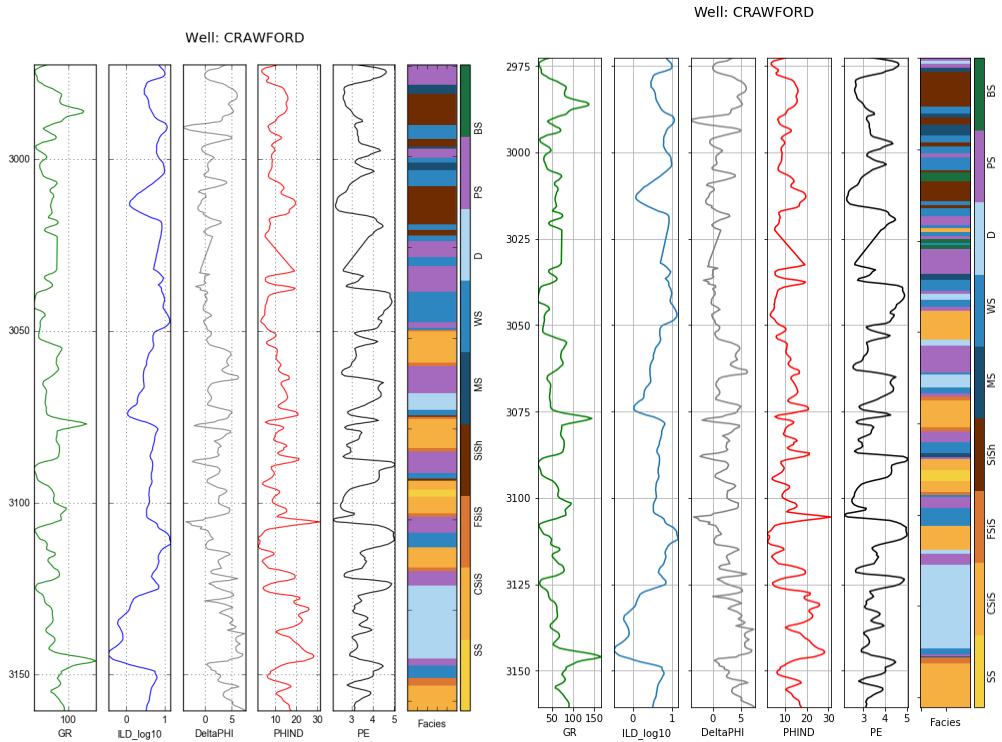


Figura 25: Poço CRAWFORD gerado por predição à esquerda e o poço verdadeiro à direita (Bestagini et al., 2017).

No poço STUART (Figura 24), o resultado gerado do modelo preditivo para um poço sem informações, para verificar a generalização, foi no geral bastante significativo, entretanto, para intercalações contidas em um tipo de rocha existente no poço verdadeiro, o modelo não obteve um desempenho de maneira satisfatória, interpretando muitas intercalações como inexistentes, e em alguns casos, interpretou a existência de intercalações em rochas que não as possuem, e os detalhes de muitas intercalações o modelo não os interpretou. Ademais, o modelo classificou de maneira incorreta partes mais rasas.

O poço CRAWFORD (Figura 25), da mesma forma do outro poço, teve dificuldades com as intercalações, classificou melhor também em profundidades maiores. De maneira geral, obteve um modelo preditivo satisfatório.

Os resultados das métricas de classificação para estes dois poços foram os seguintes conforme mostram a Tabela 4 e a matriz de confusão (Figura 26). O valor de acurácia foi de 0,46 e para fácie adjacentes, de aproximadamente 0,77.

Tabela 4: Métricas de classificação dos dados de teste (Bestagini et al., 2017).

Classes	SS	CSiS	FSiS	SiSh	MS	WS	D	PS	BS
Precisão	0,50	0,42	0,62	0,68	0,59	0,38	0,72	0,32	0,18
Recall	0,29	0,60	0,37	0,78	0,35	0,36	0,48	0,45	0,33
F1	0,36	0,49	0,46	0,73	0,44	0,37	0,58	0,37	0,24

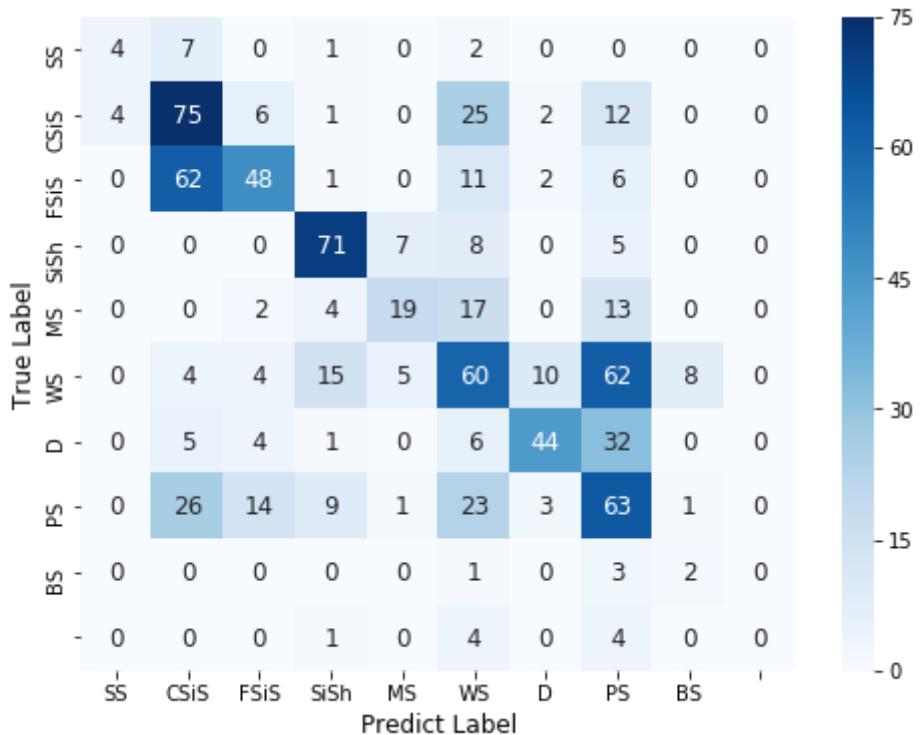


Figura 26: Matriz de confusão para os dois poços de teste STUART e CRAWFORD (Bestagini et al., 2017).

As métricas mostram que o desempenho dos dados de teste ainda está longe do ideal para um modelo preditivo que deve ser utilizado na indústria com acurácia ideal de 80% a 90%.

## 6. CONCLUSÃO

Pelos resultados obtidos, mostrou-se neste estudo que o conjunto de dados ainda não é suficiente para fazer uma análise mais profunda do estudo, necessitando de dados mais representativos de cada classe para evitar um maior viés do analista e aplicação de técnicas de balanceamento entre as classes. A forma como está apresentado o conjunto de dados mostra que realmente é necessário um pré-processamento de todas as variáveis

preditoras do problema, e um excelente tratamento vai garantir resultados bastante satisfatórios, a levar um bom tempo gasto nesta fase do estudo.

*Feature engineering/augmentation* mostrou-se fundamental na aplicação a este conjunto de dados para melhor desempenho, aumentando em 13% a acurácia de predição, como a matriz de confusão esboçou.

O algoritmo Gradient Boosting implementado forneceu um importante método de predição, atingindo um bom desempenho para o conjunto de dados; e o ajuste dos parâmetros de otimização é fundamental, pois evita o sobre-ajuste e o sub-ajuste, consequentemente, melhora o resultado.

Mesmo obtendo um baixo valor de acurácia de predição neste trabalho, já que uma pontuação de 90 a 95% de acurácia seria o ideal para dar confiança na implementação do modelo em um campo de óleo e gás em produção, é um grande início de implementação de métodos computacionais na área de geociências para automação de trabalhos que levam bastante tempo de execução, gerando um maior debate científico em áreas diferentes e, ao mesmo tempo, produz uma ligação entre a comunidade de geociências e área de ciência de dados, Aprendizado de Máquina e Institutos de Matemática e Computação.

## 7. REFERÊNCIAS

- Abu-Mostafa, Y., Ismail, M.M., Lin, H.S., 2012, Learning from Data AMLBook.com.
- Alexsandro, G. C., da P. Carlos, A. C., and Geraldo, G. N., 2017, Facies classification in well logs of the Namorado oilfield using Support Vector Machine algorithm, 15th International Congress of the Brazilian Geophysical Society & EXPOGEF, Rio de Janeiro, Brazil, 31 July-3 August 2017, 1853–1858.
- Alasadi S. A., Bhaya, W. S., 2017, College S Information of Technology, University of Babylon, Babil, Iraq.
- Bestagini, P., Lipari, V., and Tubaro, S., 2017, A machine learning approach to facies classification using well logs, SEG Technical Program Expanded Abstracts, 2137–2142.
- Bishop C. M., 2006, Pattern Recognition and Machine Learning: Springer, 1 edn.
- Byrnes, A. P., DuBois, M. K., Carr, T. R., Magnuson, M., 2001, Western Tight Gas Carbonates: Comparison of Council Grove Group, Panoma Field, Southwest Kansas and Western Low Permeability Sandstones.
- Caté, A., Perrozzi, L., Gloaguen, E., and Blouin, M., 2017, Machine learning as a tool for geologists: The Leading Edge, 36, No. 3, 215–219.
- Bjørlykke, K., 2015, Petroleum Geoscience from Sedimentary Environments to Rock Physics, Department of Geosciences, University of Oslo, Springer, 2 ed..
- Bohling, G. C., and M. Dubois, 2003, An integrated application of neural network and markov chain techniques to prediction of lithofacies from well logs: Technical Report 50, Kansas Geological Survey Open File Report.
- Breiman, L., Friedman, J., Stone, J. C., Olshen, R. A., 1984, Classification and Regression Trees, ed. Taylor & Francis, ISBN: 0412048418, 9780412048418.
- Breiman, L., 2001, Random Forests, Machine Learning, 45, 5–32, Kluwer Academic Publishers. Manufactured in The Netherlands.

- Domingos, P., 2012, A few useful things to know about machine learning. *Communications of the ACM*, 55(10):78–87.
- Dubois, M.K., Byrnes, A.P., Bohling, G.C., Seals, S.C., and Doveton, J.H., 2003a, Statistically-based lithofacies predictions for 3-D reservoir modeling: an example from the Panoma (Council Grove) Field, Hugoton Embayment, southwest Kansas, (abs.): American Assoc. of Petroleum Geologists, 2003 Annual Convention, vol. 12, p. A44; also in Kansas Geological Survey, Open-file Rept. 2003-30.
- Dubois, M.K., G.C. Bohling, A.P. Byrnes, and S.C. Seals, 2003b, Extracting Lithofacies from Digital Well Logs Using Artificial Intelligence, Panoma (Council Grove) Field, Hugoton Embayment, Southwest Kansas, (abs.): American Assoc. of Petroleum Resources, Mid-continent Section Meeting, Tulsa; also in Kansas Geological Survey, Open-file Rept. 2003-68.
- Dubois, M. K., G. C. Bohling, and S. Chakrabarti, 2005, Comparison of four approaches to a rock facies classification problem: *Computers & Geosciences*, 33, no. 5, 599–617, <http://dx.doi.org/10.1016/j.ageo.2006.08.011>.
- Duboue, P., 2020, The Art of feature engineering - essentials of machine learning, Cambridge University Press.
- Ellis, V. D., Singer, M. J., 2007, Well logging for Earth Science: Springer, second edn.
- Friedman, J. H., 1999, Greedy Function Approximation: A Gradient Boosting Machine.
- Guardo, M., 2018, Facies classification with features engineering, clustering, and gradient boosting trees: CREWES (University of Calgary) Research Report — Volume 30.
- Guozhu, D., G., Huan, L., 2018, Feature Engineering for Machine Learning and Data Analytics, Chapman & Hall/CRC data mining and knowledge discovery series.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014, Generative Adversarial Nets, Departement d'informatique et de recherche operationnelle, Université de Montreal.
- Hall, B., 2016, Facies classification using machine learning: *The Leading Edge*, 35, No. 10, 906–909.
- Hastie, T., Tibshirani, R., and Friedman, J., 2001, The elements of statistical learning - data mining, inference, and prediction: Springer, second edn.
- Haykin, S. , 2001 Redes Neurais princípios e práticas. trad. Paulo Martins Engel. - 2.ed. - Porto Alegre: Bookman.
- Hossin, M., Sulaiman, M.N., 2015, a Review on Evolution Metrics for Data Classification Valuations, *International Journal of Data Mining & Knowledge Management Process (IJDKP)* Vol.5, No.2.
- Kuhn M., Johnson K., 2019, Feature Engineering and Selection -A Practical Approach for Predictive Models, pub. Chapman & Hall/CRC Data Science Series, 1 edn.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P., 1998, Gradient-based learning applied to document recognition. *Proceedings of the IEEE*. 86. Retrieved October 7, 2016.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P., 1999, Object Recognition with Gradient-Based Learning: AT&T Shannon Lab.
- LeCun, Y.; Bengio, Y.; Hinton, G., 2015, Deep learning. *Nature*. 521(7553): 436–444.
- Liu H., Motoda H., Series: The Springer International Series in Engineering and Computer Science 453 Publisher: Springer US, ISBN: 978-1-4613-7622-4, 978-1-4615-5725-8, 1998.
- James, G., Witten, D., Hastie, T., Tibshirani, R., 2013, An Introduction to Statistical Learning with applications in R: Springer, first edition.
- Mitchell, M.T., 1997, Machine Learning: MIT Press e McGraw-Hill.
- Moore, R. C., 1949, Meaning of facies. *Mem.- Geol. Soc. Am.* 39, 1-34.
- Pilgrim, M., 2004. Dive into Python 2 ed. Nova Iorque: Apress. 413 páginas. ISBN 978-1-5905-9356-1.

- Principe J.C., Euliano, N. R., Lefebvre, W. C., Wiley, 2000, Fundamentals Through Simulations.
- Schön, H. J., 2016, Physical Properties of Rocks: Fundamentals and Principles of Petrophysics volume 65.
- Silva, A., Neto, I. L., Carrasquilla, A., Misságia, R., Ceia, M., and Archilha, N., 2014, Neural network computing for lithology prediction of carbonate- siliciclastic rocks using elastic, mineralogical and petrographic properties, 13th International Congress of the Brazilian Geophysical Society & EXPOGEF, Rio de Janeiro, Brazil, 26–29 August 2013, 1055–1058.
- Smith, T., 2018, What is the “Holy Grail” of Machine Learning in Seismic Interpretation ?, Geophysics Society of Houston.
- VanderPlas, J., 2016, Python Data Science Handbook: Essential Tools for Working with Data: O’reilly, first edition.
- Zheng A., Casari A., 2018, Publisher: O'Reilly Media, ISBN: 1491953241, 9781491953242.