

**MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
CURSO DE GRADUAÇÃO EM ENGENHARIA CARTOGRÁFICA**

**1 Ten MARÍLIA NASCIMENTO MONTEIRO
TIAGO NASCIMENTO MARCHIORE**

PLATAFORMA WEB PARA ANÁLISES DE SÉRIES TEMPORAIS

**Rio de Janeiro
2018**

INSTITUTO MILITAR DE ENGENHARIA

**1 Ten MARÍLIA NASCIMENTO MONTEIRO
TIAGO NASCIMENTO MARCHIORE**

**PLATAFORMA WEB PARA ANÁLISES DE SÉRIES
TEMPORAIS**

Projeto de Fim de Curso apresentado ao Curso de Graduação em Engenharia de Cartográfica do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Engenheiro Cartógrafo.

Orientador: Maj Francisco Roberto da Rocha Gomes - M.Sc.
Co-Orientadora: Prof^a. Raquel Aparecida Abrahão Costa e Oliveira - D.Sc.

Rio de Janeiro
2018

c2018

INSTITUTO MILITAR DE ENGENHARIA
Praça General Tibúrcio, 80 - Praia Vermelha
Rio de Janeiro - RJ CEP 22290-270

Este exemplar é de propriedade do Instituto Militar de Engenharia, que poderá incluí-lo em base de dados, armazenar em computador, microfilmar ou adotar qualquer forma de arquivamento.

É permitida a menção, reprodução parcial ou integral e a transmissão entre bibliotecas deste trabalho, sem modificação de seu texto, em qualquer meio que esteja ou venha a ser fixado, para pesquisa acadêmica, comentários e citações, desde que sem finalidade comercial e que seja feita a referência bibliográfica completa.

Os conceitos expressos neste trabalho são de responsabilidade do(s) autor(es) e do(s) orientador(es).

000.00 Monteiro, Marília Nascimento
x000x Plataforma Web para Análises de Séries Temporais /
Marília Nascimento Monteiro, Tiago Nascimento Marchiore, orientado por Maj Francisco Roberto da Rocha Gomes e Raquel Aparecida Abrahão Costa e Oliveira -
Rio de Janeiro: Instituto Militar de Engenharia, 2018.

59p.: il.

Projeto de Fim de Curso (graduação) - Instituto Militar de Engenharia, Rio de Janeiro, 2018.

1. Curso de Graduação em Engenharia Cartográfica - projeto de fim de curso. 1. Series Temporais. 2. Plataforma Online. 3. Meteorologia. I. Gomes, Maj Francisco Roberto da Rocha . II. Oliveira, Raquel Aparecida Abrahão Costa e . III. Título. IV. Instituto Militar de Engenharia.

INSTITUTO MILITAR DE ENGENHARIA

**1 Ten MARÍLIA NASCIMENTO MONTEIRO
TIAGO NASCIMENTO MARCHIORE**

**PLATAFORMA WEB PARA ANÁLISES DE SÉRIES
TEMPORAIS**

Projeto de Fim de Curso apresentado ao Curso de Graduação em Engenharia Cartográfica do Instituto Militar de Engenharia, como requisito parcial para a obtenção do título de Engenheiro Cartógrafo.

Orientador: Maj Francisco Roberto da Rocha Gomes - M.Sc.

Co-Orientadora: Prof^a. Raquel Aparecida Abrahão Costa e Oliveira - D.Sc.

Aprovado em 2 de Agosto de 2018 pela seguinte Banca Examinadora:

Maj Francisco Roberto da Rocha Gomes - M.Sc. do IME - Presidente

Prof^a. Raquel Aparecida Abrahão Costa e Oliveira - D.Sc. do IME

Cel R/1 Luís Antônio de Andrade - M.Sc. do IME

Prof^a. Heloísa Alves Silva Marques - D.Sc. do IME

Rio de Janeiro
2018

Ao Instituto Militar de Engenharia, alicerce da minha formação e aperfeiçoamento.

AGRADECIMENTOS

Agradecemos a todos os apoiadores de nossa trajetória profissional e acadêmica. Em especial, aos nossos professores e orientadores Major Roberto Gomes, Major Barreto e Professora Raquel Costa e Oliveira pelo suporte dado durante todo o curso profissional.

“I love deadlines. I like the whooshing sound they make as they fly by. ”

DOUGLAS ADAMS

SUMÁRIO

LISTA DE ILUSTRAÇÕES	8
LISTA DE SIGLAS	9
LISTA DE ABREVIATURAS	10
1 INTRODUÇÃO	13
1.1 Motivação	13
1.2 Objetivo	17
1.3 Justificativa	17
1.4 Cronograma	18
2 FUNDAMENTAÇÃO TEÓRICA	21
2.1 Características de Séries Temporais	21
2.1.1 Definições	21
2.1.2 Medidas de Dependência: Auto-Correlação e Correlação Cruzada	22
2.1.3 Séries Temporais Estacionárias	23
2.1.4 Estimativa da Correlação	25
2.2 Análise Exploratória de Dados	25
2.2.1 Filtros Lineares	26
2.2.2 Decomposição de Séries Temporais	27
2.3 Processos Estacionários	27
2.3.1 Processos Autorregressivos (AR)	27
2.3.2 Processos de Média Móvel (MA)	28
2.3.3 Processos Autorregressivos de Média Móvel (ARMA)	29
2.4 Processos não-estacionários	29
2.4.1 Processos Autorregressivos Integrados de Média Móvel (ARIMA)	29
3 REQUISITOS DE PROJETO	30
3.1 Estrutura da Plataforma	30
3.2 Algoritmos	31
3.3 Formatos de Entrada de Dados	31
4 METODOLOGIA	33
4.1 Ferramentas Empregadas	33

4.1.1	Desenvolvimento do Código	33
4.1.2	Gerenciamento do Projeto	33
4.2	Páginas Estáticas	34
4.3	Estrutura dos Algoritmos	35
4.4	Estrutura da Plataforma	40
4.4.1	Adição de novas páginas	41
4.4.2	Adição de novos algoritmos	42
5	PLATAFORMA <i>WEB</i> (VERSÃO 1.0)	45
5.1	Página Inicial	45
5.2	Documentação	46
5.3	Formato de Arquivos	46
5.4	Exemplos	47
5.5	Algoritmos	48
5.5.1	Função de Autocorrelação	49
5.5.2	Função de Autocorrelação Parcial	50
5.5.3	Decomposição de Séries	51
5.5.4	Média Móvel	53
6	REFERÊNCIAS BIBLIOGRÁFICAS	55
7	APÊNDICES	58
7.1	APÊNDICE 1: Bibliotecas Empregadas	59

LISTA DE ILUSTRAÇÕES

FIG.1.1	Exemplo de Monitoramento dos Deslocamentos Verticais do nível dos mares (DIDOVA et al., 2016)	15
FIG.1.2	Séries Temporais de Precipitação e Evapotranspiração Mensal em Berlin, 1851-2010 (GOSSEL; LAEHNE, 2013)	16
FIG.1.3	Cronograma de Projeto	19
FIG.3.1	Esquema Tela Inicial Plataforma	31
FIG.4.1	Repositório do Projeto no GitHub	34
FIG.4.2	Página de Apresentação dos Algoritmos	36
FIG.4.3	Exemplo de formulário para envio de informações	36
FIG.4.4	Exemplo de página para exibição de resultados	37
FIG.5.1	Página Inicial - Versão 1.0	45
FIG.5.2	Documentação - Versão 1.0	46
FIG.5.3	Formato de Arquivos - Versão 1.0	47
FIG.5.4	Exemplos de dados - Versão 1.0	47
FIG.5.5	Exemplos, download dos arquivos	48
FIG.5.6	Algoritmos - Versão 1.0	49
FIG.5.7	Função de Autocorrelação - Versão 1.0	50
FIG.5.8	Resultados da Função de Autocorrelação - Versão 1.0	50
FIG.5.9	Função de Autocorrelação Parcial - Versão 1.0	51
FIG.5.10	Resultados da Função de Autocorrelação Parcial - Versão 1.0	51
FIG.5.11	Decomposição de Séries - Versão 1.0	52
FIG.5.12	Resultados da Decomposição de Séries - Versão 1.0	53
FIG.5.13	Função de Média Móvel - Versão 1.0	54
FIG.5.14	Resultados da Função de Média Móvel - Versão 1.0	54

LISTA DE SIGLAS

GPS	Global Positioning System
GRACE	Gravity Recovery and Climate Experience
CSV	Comma Separated File

LISTA DE ABREVIATURAS

ABREVIATURAS

iid	-	Independente e Identicamente Distribuído
MA	-	<i>Moving Average</i>
AR	-	<i>Autoregressive</i>
ARMA	-	<i>Autoregressive Moving Average</i>
ARIMA	-	<i>Autoregressive Integrated Moving Average</i>
ACF	-	<i>Autocorrelation Function</i>
PACF	-	<i>Partial Autocorrelation Function</i>

SÍMBOLOS

μ	-	média
γ	-	covariância
ρ	-	correlação
P	-	distribuição conjunta de probabilidade

RESUMO

O relatório de projeto apresenta as etapas de desenvolvimento de uma plataforma online de código aberto para a realização de análises de séries temporais visando aplicações geoespaciais. Os métodos disponíveis na plataforma bem como os resultados provenientes de suas implementações são demonstrados por meio de embasamento teórico sucinto e testes dos mesmos empregando dados geoespaciais. É descrita a estrutura empregada na implementação da plataforma e o resultado final do sistema desenvolvido. A plataforma foi construída na linguagem de programação *Python* empregando também o *Framework Flask*, e os algoritmos implementados a partir da biblioteca *statsmodels*.

ABSTRACT

The current project report presents the development steps employed in order to build an online based platform with open-source designed to perform time series analysis focusing on geospatial data. The methods available at the platform as well as the results provenient from those methods are shown through a short theoretical description followed by test results employing real data on the platform methods. The coding struture is broadly presented and the website results (layouts and tests) are also shown along the report. The system was built using the programming language Python and the framework Flask, and the algorithms implemented using available package *statsmodels*.

1 INTRODUÇÃO

A aplicação de métodos de análise e modelos estatísticos para a extração de informações de séries de dados temporais é amplamente empregada tanto na área científica quanto para a geração de produtos comerciais. Para fins desse texto, uma série temporal é definida como um conjunto de observações X_t , cada uma sendo obtida num específico momento no tempo, t (BROCKWELL; DAVIS, 2002).

A análise de séries temporais tem aplicações em diversas áreas da Cartografia, não somente com dados unidimensionais mas abrangendo também dados que consideram a informação espacial dos dados (dados geo-localizados). Os métodos e modelos empregados para a performance das análises são amplamente conhecidos, mas enxergam-se limitações quanto à disponibilidade de soluções computacionais gratuitas para fins acadêmicos que solucionem de maneira eficiente problemas envolvendo séries temporais.

O presente trabalho se propõe nesse sentido a apresentar uma solução online e gratuita para a realização de análises de séries de dados temporais, com foco em aplicações espaciais.

1.1 MOTIVAÇÃO

A Análise de Séries Temporais ainda que seja uma área em constante desenvolvimento, não é tida como uma barreira de pesquisa, ou seja, objeto sobre o qual é necessário o desenvolvimento de novos modelos matemáticos capazes de solucionar quaisquer demandas existentes; pelo contrário, os modelos estatísticos conhecidos, geram resultados satisfatórios.

A seguir são apresentadas aplicações possíveis advindas da exploração de séries de dados temporais na área da Engenharia Cartográfica empregando modelos estatísticos e matemáticos existentes. É aqui realizada, apenas para fins de organização do texto, uma segmentação das aplicações nas seguintes áreas de conhecimento relacionadas à Engenharia Cartográfica: Sensoriamento Remoto, Geodésia e Sistemas de Informação Geoespacial.

Na área de Sensoriamento Remoto, são vislumbradas aplicações tanto envolvendo análises de séries de imagens temporalmente distribuídas, como análises de dados extraídos de conjuntos de imagens; a distinção entre as duas categorias se configura basicamente na consideração ou não da variável espacial nas imagens. Dentre as possíveis aplicações

podem-se citar:

- Análise de Ecossistemas Marinhos com base na resposta espectral de oceanos: a potencial aplicação se enquadra no uso de informações unidimensionais a partir de dados de sensoriamento remoto, e foi desenvolvida por pesquisadores da Espanha (CORDEIRO et al., 2016a). Trata-se da análise da variação da resposta espectral de uma amostra do oceano da região de estudo ao longo do tempo, a partir da qual são feitos estudos dos ecossistemas;
- Avaliação de Catástrofes Naturais: trata-se do mapeamento de furacões, terremotos, entre outros, através de imagens de sensores remotos, com o objetivo de analisar o impacto de tais desastres e assim determinar estratégias pré e pós seus acontecimentos (CORDEIRO et al., 2016b);
- Gerenciamento de Recursos Naturais: pode-se empregar o mapeamento de uso da terra, de regiões com presença de água e do desenvolvimento habitats para analisar temporalmente o impacto do crescimento urbano (CORDEIRO et al., 2016b);
- Controle de regiões costeiras: com séries temporais de imagens de regiões costeiras pode-se analisar o transporte de sedimentos e com isso atuar no controle de erosões.

Na área de Geodésia, são observadas aplicações variadas de séries temporais, que vão desde o monitoramento de coordenadas para a estudo de fenômenos relacionados à variação das mesmas, até a análise temporal de parâmetros geofísicos da Terra. Dentre as potenciais aplicações pode-se citar:

- Uso de Dados GPS (*Global Positioning System*) e GRACE (*Gravity Recovery and Climate Experience*) para monitoramento do nível dos mares: a aplicação consiste no uso de dados de gravidade, advindos da missão GRACE, e de posição, obtidos através dos satélites da constelação GPS, para uma estação na antártica, de modo a melhorar a estimativa do nível dos mares analisado enquanto séries de dados com uma taxa de variação do no tempo. A Figura 1.1 apresenta um exemplo da Série Temporal. A análise pode ser realizada de forma tradicional, com o uso de ajustamento dos mínimos quadrados, ou usando uma abordagem diferenciada com filtro de Kalman (DIDOVA et al., 2016);
- Análise de séries temporais de coordenadas de rede geodésica: podem ser construídas análises das coordenadas tridimensionais de estações pertencentes a uma rede geo-

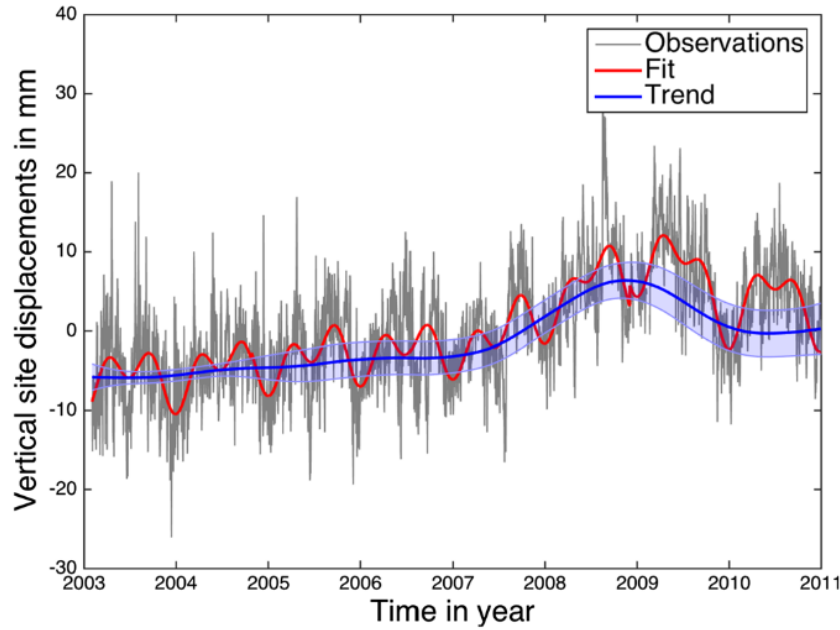


FIG. 1.1: Exemplo de Monitoramento dos Deslocamentos Verticais do nível dos mares (DIDOVA et al., 2016)

désica de forma a prever a deformação dessa rede ao longo do tempo (VELSINK, 2016);

- Análise de séries temporais de coordenadas de rede geodésica: podem ser construídas análises das coordenadas tridimensionais de estações pertencentes a uma rede geodésica de forma a prever a deformação dessa rede ao longo do tempo (VELSINK, 2016);
- Estudo das relações entre os sistemas terrestres e suas causas geofísicas: dados de geometria da Terra, parâmetros de orientação e movimentação (rotação, nutação, translação, entre outros) e informações de gravidade são constantemente coletadas e armazenadas na forma de séries temporais. A análise temporal desses dados de forma conjunta permite que sejam estabelecidas relações entre os diversos fenômenos, bem como suas causas (KOSEK, 2018).

Explorando a área de Informações Geoespaciais, distinguem-se as análises de séries temporais que consideram a variável espacial na análise, e aquelas que empregam dados como univariados (considera-se apenas uma única informação na série de dados além do tempo). São, assim, potenciais aplicações:

- Meteorologia e Climatologia: dados meteorológicos e climatológicos podem ser analisados enquanto séries temporais de forma a estabelecer análises de longo prazo

e fazer previsões (GOSSEL; LAEHNE, 2013). Trata-se, nesse sentido, de um problema de dados unidimensionais quando tratado de forma individualizada para uma estação ou região. São exemplos de dados passíveis de tal tratamento: índices pluviométricos, umidade do ar, ventos, temperatura, nível médio dos mares e avanço das marés. A Figura 1.2 apresenta dados temporais de precipitação e evapotranspiração, exemplificando a aplicação.

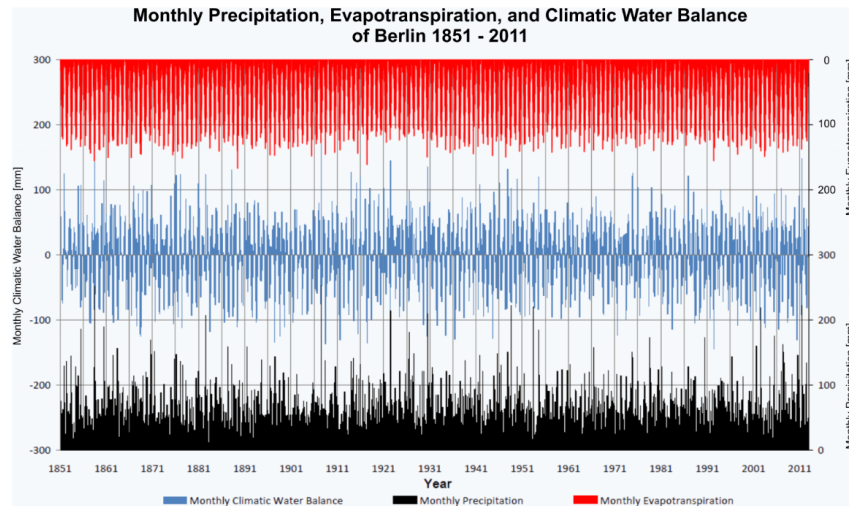


FIG. 1.2: Séries Temporais de Precipitação e Evapotranspiração Mensal em Berlim, 1851-2010 (GOSSEL; LAEHNE, 2013)

- Hidrografia: análise e previsão do regime hidrológico em águas subterrâneas e superficiais com base em longas séries de medições (GOSSEL; LAEHNE, 2013). Trata-se também de um caso de dado unidimensional, mas cuja interpretação é diretamente extrapolada no espaço;
- Dados Especiais: entende-se dados espaciais como informações coletadas atribuídas a um ponto ou região geo-localizados, como por exemplo: dados demográficos, informações de compra, distribuição de renda, consumo de serviços, valoração de imóveis, etc. A análise de tais dados na forma de série temporal considerando a relação espacial entre as observações (LEONARD; SAMY, 2015) é também uma potencial aplicação de séries temporais com dados geográficos, não se tratando, nesse caso, da análise de dados de forma unidimensional, como nos casos apresentados anteriormente.

Com base nas informações apresentadas anteriormente, fica claro o potencial de aplicação de séries temporais nas mais diversas áreas exploradas para a geração de informações. A performance de análises dos dados é por vezes subutilizada nas geociências devido

à restrição de plataformas para a geração de tais análises e ao comum alto custo atrelado à sua utilização.

Uma solução alternativa à implementação dos métodos seria o emprego de plataformas gratuitas que possuísem confiabilidade para fins acadêmicos de modo a gerar análises de forma rápida e eficiente.

1.2 OBJETIVO

Com base nas motivações apresentadas, o presente trabalho tem por objetivo:

Desenvolver uma plataforma online e aberta para a resolução de problemas envolvendo séries temporais, através de alguns dos principais algoritmos matemáticos e métodos de análise consagrados.

As decisões de desenvolvimento do projeto envolvendo quais algoritmos ou métodos serão implementados, e quais os requisitos técnicos da plataforma serão discutidos em detalhes nos Capítulos a seguir.

1.3 JUSTIFICATIVA

A plataforma a ser desenvolvida precisa seguir algumas diretrizes, definidas com base no exposto na Seção 1.1:

- Gratuita, para a comunidade acadêmica;
- Intuitiva, ou seja, de fácil utilização e interpretação;
- Implementada em língua portuguesa; e
- Open-source, de modo a incentivar melhorias contínuas.

A decisão pela construção de uma plataforma própria é explicada ao se observar algumas das soluções atualmente existentes, verificando-se que não é possível atender integralmente às diretrizes com apenas uma solução disponível, mas sim com uma combinação entre várias.

Pode-se dividir as soluções encontradas em três categorias:

- 1 Softwares estatísticos comerciais;
- 2 Plataformas open-source;

3 Soluções online;

Entre os principais softwares comerciais, destacam-se o Stata©, amplamente utilizado por economistas, o SAS®, de maior utilização entre os estatísticos e o MATLAB®. Todos os softwares apresentam soluções completas para séries temporais, vide (STATA, 2017) e (BROCKLEBANK; DICKEY, 2003). As principais vantagens de uma solução comercial são a abrangência de soluções do produto, o melhoramento contínuo pelos desenvolvedores e a facilidade em retirada de dúvidas através de documentação e participação da comunidade, que contém milhares de usuários ativos. As desvantagens, porém, são os altos custos de licença e manutenção, além do usuário não ter acesso ao código-fonte.

Nas plataformas open-source, destaca-se o R, que consiste na maior linguagem e ambiente para análises estatísticas e gráficos. Sua descrição e proposta encontra-se em (R, 2018). Suas principais vantagens são a disponibilização de todos os códigos-fonte (e possível adição no ambiente de implementações autorais), a facilidade em retirada de dúvidas, através da comunidade e documentação, e o fato de ser gratuito. A principal desvantagem é a difícil curva de aprendizado para um usuário iniciante, sendo necessário tempo de estudo e familiaridade antes que se possa resolver diretamente o problema pensado.

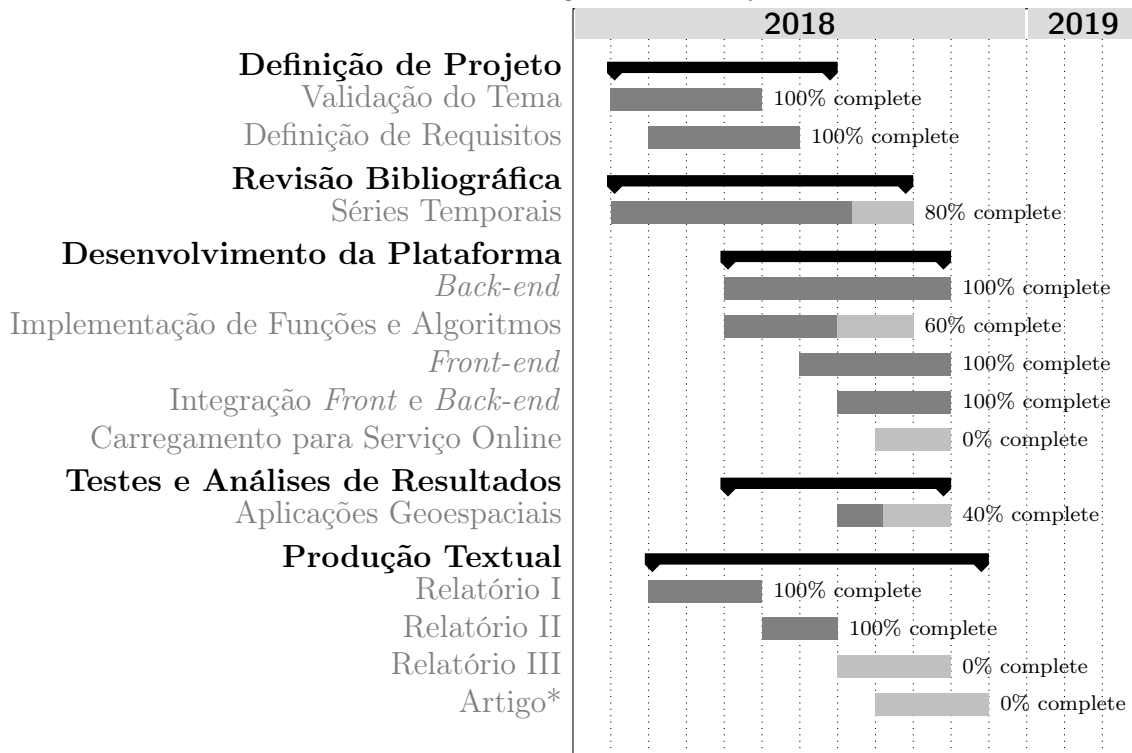
Por fim, nas soluções online existem alguns sites acadêmicos disponíveis, e até mesmo soluções pagas em nuvem. Destacam-se: Microsoft Azure "Time Series Insights", que é uma solução paga para processamento de dados em nuvem e Machine Learning, e que inclui soluções de análise de séries temporais, desenvolvido pela Microsoft; e Wessa (WESSA, 2018), uma solução acadêmica online onde é possível escolher o algoritmo a ser utilizado, determinar parâmetros e dados, e interpretar os resultados na própria página web. Dentre as vantagens das soluções encontradas nessa categoria, destacam-se a facilidade e agilidade de uso, dado que não há necessidade de instalação ou preparação de ambiente, e os resultados são disponibilizados na própria página. As desvantagens são específicas às soluções encontradas: as implementações são em inglês, e algumas soluções podem ser pagas e/ou incompletas.

Portanto, dada a natureza de cada uma das categorias de soluções disponíveis, determinou-se o objetivo do projeto e os requisitos principais.

1.4 CRONOGRAMA

Segue na Figura 1.3 o cronograma com as atividades previstas para o desenvolvimento do Projeto. Considera-se como período útil à realização do projeto de 05 de fevereiro de

FIG. 1.3: Cronograma de Projeto



2018 a 05 de outubro de 2018.

São descritos a seguir de forma breve as atividades pretendidas para cada um dos itens descritos no cronograma apresentado:

1 Definição de Projeto

- (a) Validação do Tema: a etapa visa a validação da proposta do projeto, com a busca com ferramentas existentes comparáveis ou iguais ao que se planeja desenvolver, elaboração de comparações e ainda a análise de aplicações no contexto da Engenharia Cartográfica.
- (b) Definição de Requisitos: definição dos requisitos básicos da plataforma, premissas que deverão ser cumpridas até o fim do projeto.

2 Revisão Bibliográfica

- (a) Séries Temporais: revisão bibliográfica de séries temporais, abrangendo conceitos básicos mas com foco nos métodos de análise para futura implementação.

3 Desenvolvimento da Plataforma

- (a) *Back-end*: construção do sistema onde estarão embarcados os algoritmos e processos de gerenciamento e leitura de dados.

- (b) Implementação de Funções e Algoritmos:
- (c) *Front-end*: construção da interface gráfica da plataforma atendendo aos requisitos de projeto estabelecidos.
- (d) Integração *Front-end* e *Back-end*: integração entre a interface gráfica e o sistema construído.
- (e) Carregamento para Serviço Online: carregamento da plataforma para serviço online de forma a viabilizar seu acesso aberto na internet.

4 Testes e Análises de Resultados

- (a) Outras Aplicações Geoespaciais: utilização de séries temporais da área geoespacial com temas diversos empregando os recursos disponíveis na plataforma.

5 Produção Textual

- (a) Relatórios: produção de relatórios parcial (I e II) e final (III).
- (b) Artigo* (opcional): produção de artigo acerca do tema desenvolvido ao longo do projeto.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 CARACTERÍSTICAS DE SÉRIES TEMPORAIS

2.1.1 DEFINIÇÕES

Segundo (BROCKWELL; DAVIS, 2002), uma série temporal pode ser classificada em duas possíveis categorias:

- Série Temporal Discreta: o conjunto T_0 de tempos em que as observações são realizadas é um conjunto discreto, como por exemplo, no caso de observações realizadas em intervalos fixos.
- Série Temporal Contínua: obtidas quando as observações são obtidas continuamente em um determinado intervalo de tempo, como por exemplo no caso em que o dito conjunto T_0 acima citado é da forma $T_0 = [0, 1]$.

Distinguem-se duas abordagens para a Análise de Séries Temporais (SHUMWAY; STOFFER, 2017):

- Domínio do Tempo: em geral motivada pela suposição de que a correlação entre pontos no tempo é melhor explicada em termos da dependência entre o valor atual e valores passados, ou seja, seja um evento x_t ocorrido no tempo t , ele depende de alguma forma de eventos anteriores x_{t-1}, x_{t-2}, \dots (SHUMWAY; STOFFER, 2017). A abordagem do domínio do tempo é focada na modelagem de valores futuros de uma série temporal como sendo uma **função paramétrica** de valores atuais.
- Domínio da Frequência: assume que o fenômeno temporal de interesse primário para a análise de séries temporais se relaciona com alguma **variação senoidal periódica ou sistemática** naturalmente encontrada na maioria dos dados, interpretando a Série Temporal não como um processo estocástico (domínio do tempo), mas como um sinal.

O presente trabalho, nesse sentido tem por foco a análise e construção de modelos empregando Séries Temporais Discretas.

Alguns padrões de comportamento comumente encontrados em séries temporais (FERREIRA, 2018), são definidos abaixo:

- Tendência: comportamento observado quando a série temporal seguir uma determinada direção, crescente ou decrescente, e não necessariamente linear.
- Sazonalidade: padrão que se repete com uma certa periodicidade dentro de um ano.
- Ciclo: padrão que se repete com alguma regularidade, mas sem período fixo e dentro de um período maior que 1 ano.

2.1.2 MEDIDAS DE DEPENDÊNCIA: AUTO-CORRELAÇÃO E CORRELAÇÃO CRUZADA

Seguem abaixo definições de medidas descritivas de informações marginais de séries temporais, isto é, medidas estatísticas capazes de descrever, em algum grau, características de uma série temporal (SHUMWAY; STOFFER, 2017).

Definição 1. *A função média é definida como:*

$$\mu_{xt} = E(x_t) = \int_{-\infty}^{\infty} x f_t(x) dx, \quad (2.1)$$

onde E denota o operador de valor esperado ou esperança. A notação μ_{xt} poderá ser substituída pela notação μ_t desde que no contexto não haja perda de clareza.

Definição 2. *A autocovariância é definida como:*

$$\gamma_x(s, t) = cov(x_s, x_t) = E[(x_s - \mu_s)(x_t - \mu_t)] \forall s, t, \quad (2.2)$$

onde s e t denotam pontos no tempo onde amostras x são observadas. Na equação acima a notação $\gamma_x(s, t)$ poderá ser substituída pela notação $\gamma(s, t)$ desde que não haja perda de clareza no contexto em que for empregada (BROCKWELL; DAVIS, 2002).

A autocovariância mede a dependência linear entre dois pontos numa mesma série observados em momentos diferentes na escala de tempo.

Definição 3. *A função de autocorrelação é definida como (SHUMWAY; STOFFER, 2017):*

$$\rho(s, t) = \frac{\gamma(s, t)}{\sqrt{\gamma(s, s)\gamma(t, t)}}. \quad (2.3)$$

A função de autocorrelação mede a predizibilidade de uma série num ponto t , x_t , usando unicamente o valor do ponto s , x_s . Demonstra-se que $-1 \leq \rho(s, t) \leq 1$. Valores

de $|\rho(s, t)|$ próximos de 1, indicam que x_t pode ser explicado por x_s através de uma relação linear, sendo essa explicação perfeita para $|\rho(s, t)| = 1$.

Cabe notar que a definição de autocorrelação avalia a relação entre duas amostras no tempo separada por um intervalo, e portanto, o intervalo de amostras entre os pontos influencia e adiciona um viés ao dado de autocorrelação apresentado. Surge, assim, como forma de evitar o efeito dos dados do intervalo, o conceito de autocorrelação parcial.

Definição 4. A autocorrelação parcial (*partial autocorrelation function (PACF)*) de um processo estacionário, x_t , denotada por ϕ_h^h ou por ϕ_{hh} para $h = 1, 2, \dots$ (NOSEDAL, 2016)

$$\phi_{hh} = \text{corr}(x_{t+h} - \hat{x}_{t+h}), h \geq 2. \quad (2.4)$$

Definição 5. A covariância cruzada entre duas séries, x_t e y_t é dada por (SHUMWAY; STOFFER, 2017):

$$\gamma_{xy}(s, t) = \text{cov}(x_s, y_t) = E[(x_s - \mu_{xs})(y_t - \mu_{yt})]. \quad (2.5)$$

Definição 6. A correlação cruzada entre duas séries, x_t e y_t é dada por (SHUMWAY; STOFFER, 2017):

$$\rho_{xy}(s, t) = \frac{\gamma_{xy}(s, t)}{\sqrt{\gamma_x(s, s)\gamma_y(t, t)}}. \quad (2.6)$$

2.1.3 SÉRIES TEMPORAIS ESTACIONÁRIAS

Uma série temporal é estacionária quando sua distribuição de probabilidade é estável ao longo do tempo, ou seja, se tomarmos um conjunto aleatório de variáveis na sequência e deslocarmos em h períodos de tempo, a distribuição de probabilidade conjunta deve permanecer inalterada. A definição formal é apresentada a seguir.

Definição 7. Uma série temporal *estritamente estacionária* é aquela para a qual o comportamento probabilístico para toda coleção de valores:

$$\{x_{t_1}, x_{t_2}, \dots, x_{t_k}\} \quad (2.7)$$

é idêntico ao daquele de um conjunto deslocado no tempo:

$$\{x_{t_1+h}, x_{t_2+h}, \dots, x_{t_k+h}\} \quad (2.8)$$

ou seja,

$$P\{x_{t_1}, x_{t_2}, \dots, x_{t_k}\} = P\{x_{t_1+h}, x_{t_2+h}, \dots, x_{t_k+h}\}, \quad (2.9)$$

para todo $k = 1, 2, \dots$, todo ponto no tempo t_1, t_2, \dots, t_k , toda constante c_1, c_2, \dots, c_k e todos os deslocamentos no tempo $h = 0, \pm 1, \pm 2, \dots$ (SHUMWAY; STOFFER, 2017). Em 2.9 P é a distribuição de probabilidade conjunta.

São propriedades de uma série temporal estritamente estacionária X_t (BROCKWELL; DAVIS, 2002):

- 1 As variáveis aleatórias x_t são identicamente distribuídas;
- 2 $P\{x_t, x_{t+h}\} = P\{x_1, x_{1+h}\} \forall t, h \in \mathbb{Z}$;
- 3 $E[x_t] \neq f(t)$, onde $f(t)$ é uma função do tempo;
- 4 $\gamma(t, t+h) = \gamma(1, 1+h)$ e $\gamma(t, t+h) \neq f(t)$ onde $f(t)$ é uma função do tempo;
- 5 Uma sequência *iid* (independente e identicamente distribuída) é estritamente estacionária.

A definição de estacionaridade apresentada na Definição 7 é muito rígida para a maioria das aplicações; segue, portanto, o conceito de uma série temporal fracamente estacionária.

Definição 8. Uma série temporal fracamente estacionária, x_t , é um processo de variância finita tal que (SHUMWAY; STOFFER, 2017):

- (i) a função de valor médio, μ_t , é constante e não depende do tempo;
- (ii) a função de autocovariância, $\gamma(s, t)$ depende de s e t apenas através de sua diferença $|s - t|$

Ao longo do texto o termo **série estacionária** poderá ser utilizado no lugar de **série fracamente estacionária**.

Quando diversas séries estão disponíveis, a noção de estacionaridade ainda se aplica com condições adicionais.

Definição 9. Sejam x_t e y_t duas séries temporais; elas serão ditas **conjuntamente estacionárias** se cada uma delas é estacionária e se a função de covariância cruzada (SHUMWAY; STOFFER, 2017):

$$\gamma_{xy}(h) = \text{cov}(x_{t+h}, y_t) = E[(x_{t+h} - \mu_x)(y_t - \mu_y)] \quad (2.10)$$

é função somente do passo h .

2.1.4 ESTIMATIVA DA CORRELAÇÃO

Os modelos de estacionariedade apresentados na Sub-seção 2.1.3 são de extrema utilidade para a definição teórica de modelos aplicados a séries hipotéticas. Com conjuntos discretos de dados, porém, a limitação de amostras permite apenas que se estimem os valores da média, variância, autocovariância e autocorrelação das séries, e além disso a suposição de que as amostras são independentes e identicamente distribuídas (*iid*) é impossível na maior parte dos casos. São apresentados a seguir as definições para o cálculo das estimativas dos parâmetros apresentados nas subseções anteriores de forma a viabilizar o estudo envolvendo conjuntos de dados discretos.

Definição 10. *A variância de uma amostra é definida por (SHUMWAY; STOFFER, 2017):*

$$\hat{\gamma}(h) = \frac{1}{n} \sum_{t=1}^{n-h} (x_{t+h} - \bar{x})(x_t - \bar{x}), \quad (2.11)$$

com $\hat{\gamma}(-h) = \hat{\gamma}(h)$ para $h = 0, 1, \dots, n-1$, onde:

$$\bar{x} = \frac{1}{n} \sum_{t=1}^n x_t$$

Definição 11. *A função de autocorrelação para uma amostra é definida por (FALK et al., 2005):*

$$\hat{\rho}(h) = \frac{\hat{\rho}(h)}{\hat{\rho}(0)}, h = 1, 2, 3... \quad (2.12)$$

Definição 12. *Os estimadores para $\gamma_{xy}(h)$ e $\rho_{xy}(h)$ são dados, respectivamente pela função de covariância cruzada para amostras (SHUMWAY; STOFFER, 2017):*

$$\hat{\gamma}_{xy}(h) = \frac{1}{n} \sum_{t=1}^{n-h} (x_{t+h} - \bar{x})(y_t - \bar{y}), \quad (2.13)$$

com $\hat{\gamma}_{xy}(-h) = \hat{\gamma}_{xy}(h)$ determinando a função para valores de h negativos, e:

$$\hat{\rho}_{xy}(h) = \frac{\hat{\rho}_{xy}(h)}{\sqrt{\hat{\rho}_x(0)\hat{\rho}_y(0)}} \quad (2.14)$$

2.2 ANÁLISE EXPLORATÓRIA DE DADOS

Em dados de séries temporais a dependência entre os valores de uma série é uma medida fundamental, dessa forma as estimativas dos valores de autocorrelação devem ser precisos. Se a dependência entre os valores varia ponto a ponto, a análise dessa dependência se torna inviável. Além disso, para que se atinja uma análise estatística significativa dos dados

da série temporal é necessário que as funções de média e autocovariância satisfaçam à condição de estacionaridade. Ocorre porém que na maioria dos casos as ditas condições não são atendidas. A presente seção tem por objetivo, portanto, apresentar métodos que auxiliem na conversão dos dados de séries temporais para formatos que permitam o estudo das propriedades de estacionaridade.

Definição 13. *Define-se o operador backshift por (SHUMWAY; STOFFER, 2017):*

$$Bx_t = x_{t-1} \quad (2.15)$$

e estendendo a definição para potências maiores, como por exemplo: $B^2x_t = B(Bx_t) = Bx_{t-1} = x_{t-2}$, segue por definição que:

$$B^k x_t = x_{t-k}, \quad (2.16)$$

onde B é uma constante.

Definição 14. *Define-se a **diferença de ordem d** como sendo (SHUMWAY; STOFFER, 2017):*

$$\nabla^d = (1 - B)^d \quad (2.17)$$

Se $d = 1$ ele poderá ser omitido da notação.

A primeira diferença (ou diferença de ordem 1) elimina a tendência linear; a segunda diferença elimina a tendência quadrática, e assim por diante.

2.2.1 FILTROS LINEARES

A utilização de filtros lineares possui diversas aplicações no contexto de Séries Temporais, sendo um processo auxiliar na **Suavização de Dados** e detecção de tendências na série, por exemplo.

Define-se de forma geral que um processo $\{Y_t\}$ é o resultado de um filtro linear $C = \{c_{t,k}, k = 0, \pm 1, \dots\}$ aplicado à série temporal original $\{X_t\}$ se (BROCKWELL; DAVIS, 2002):

$$Y_t = \sum_{j=-\infty}^{\infty} c_{t,k} X_k, t = 0, \pm 1, \dots \quad (2.18)$$

E tal filtro é dito **invariante no tempo** se os pesos $c_{t,t-k}$ forem independentes de t , isto é, se $c_{t,t-k} = \psi_k$ (BROCKWELL; DAVIS, 2002).

Dentre os possíveis filtros lineares suavizadores, a média móvel é o mais comumente utilizado, podendo ter sua referência centrada (simétrica), ou à direita (ou seja, utilizando

apenas dados passados). Média móvel implica que todos os pesos são iguais em todos os pontos de dados no intervalo de aplicação do filtro.

2.2.2 DECOMPOSIÇÃO DE SÉRIES TEMPORAIS

Um dos objetivos em realizar análise exploratória dos dados de Séries Temporais é encontrar padrões de repetição nos dados. Para isso, pode-se utilizar métodos de decomposição de séries temporais.

Entre os métodos consagrados encontra-se o Modelo de Decomposição Clássica (BROCKWELL; DAVIS, 2002), que divide a série em três componentes: a primeira componente refere-se à **tendência** (*trend component*), que consiste em uma série de mudanças lentas; a segunda refere-se à **sazonalidade** (*seasonal component*), ou seja, uma série com período conhecido e fixo; a terceira refere-se ao **resíduo** (*random noise component*), que consiste em uma série estacionária.

O Modelo de Decomposição pode ser do tipo Aditivo ou Multiplicativo. Considerando a Série Temporal original como X_t , e as componentes de tendência, sazonalidade e resíduo como, respectivamente, m_t , s_t e Y_t , tem-se:

Modelo aditivo:

$$X_t = m_t + s_t + Y_t \quad (2.19)$$

Modelo multiplicativo:

$$X_t = m_t \cdot s_t \cdot Y_t \quad (2.20)$$

Para a estimação da componente de tendência, há diferentes abordagens possíveis (BROCKWELL; DAVIS, 2002): pode-se utilizar um Filtro de Suavização, apresentado na seção 2.2.1, ou estimação de polinômio que melhor se ajuste aos pontos da série.

A partir da estimação, realiza-se o processo de remoção de tendência (*detrend*) nos dados originais (seja por subtração, no caso do Modelo Aditivo, como por divisão, no caso do Modelo Multiplicativo). Com a nova série resultante, extrai-se as variáveis de sazonalidade e resíduo.

2.3 PROCESSOS ESTACIONÁRIOS

2.3.1 PROCESSOS AUTORREGRESSIVOS (AR)

Modelos autorregressivos são baseados na ideia de que o valor atual de uma série, x_t , pode ser explicado como uma função de p valores passados, $x_{t-1}, x_{t-2}, \dots, x_{t-p}$, onde p

representa o número de passos em direção ao passado necessários para prever o valor presente (SHUMWAY; STOFFER, 2017).

Definição 15. Um *modelo autorregressivo* (autoregressive model) de ordem p abreviado na forma $AR(p)$, é da forma (SHUMWAY; STOFFER, 2017):

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \omega_t \quad (2.21)$$

onde, x_t é estacionária, e $\phi_1, \phi_2, \dots, \phi_p$ são constantes, com $\phi_p \neq 0$; ω_t é um ruído branco gaussiano com média zero e variância σ_ω^2 , a não ser que seja definido outro tipo de ruído para essa variável de acordo com as condições do problema.

A média de x_t é zero. Se a média μ de x_t não for originalmente zero deve-se substituir x_t por $x_t - \mu$ em 2.21, de onde vem:

$$x_t - \mu = \phi_1(x_{t-1} - \mu) + \phi_2(x_{t-2} - \mu) + \cdots + \phi_p(x_{t-p} - \mu) + \omega_t \quad (2.22)$$

ou ainda:

$$x_t = \alpha + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \omega_t, \quad (2.23)$$

onde, $\alpha = \mu(1 - \phi_1 - \phi_2 - \cdots - \phi_p)$.

Definição 16. O *operador autorregressivo* é definido como (SHUMWAY; STOFFER, 2017):

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p \quad (2.24)$$

A **condição de estacionariedade** de um processo $AR(p)$ é dada por (FALK et al., 2005):

$$1 - a_1 z - a_2 z^2 - \cdots - a_p z^p, |z| \leq 1 \quad (2.25)$$

2.3.2 PROCESSOS DE MÉDIA MÓVEL (MA)

Definição 17. Um *modelo de média móvel* (moving average model) de ordem q abreviado na forma $MA(q)$, é da forma (SHUMWAY; STOFFER, 2017):

$$x_t = \omega_t + \theta_1 \omega_{t-1} + \theta_2 \omega_{t-2} + \cdots + \theta_q \omega_{t-q} \quad (2.26)$$

onde x_t é estacionário, e existem q atrasos na média móvel e $\theta_1, \theta_2, \dots, \theta_q$ ($\theta_q \neq 0$) são os parâmetros; ω_t é um ruído branco gaussiano com média zero e variância σ_ω^2 , a não ser que seja definido outro tipo de ruído para essa variável de acordo com as condições do problema.

Definição 18. O *operador de média móvel* é definido como (SHUMWAY; STOFFER, 2017):

$$\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q. \quad (2.27)$$

2.3.3 PROCESSOS AUTORREGRESSIVOS DE MÉDIA MÓVEL (ARMA)

A partir das definições 15 e 17, define-se um modelo misto para séries temporais estacionárias, abreviado na forma *ARMA*.

Definição 19. *Uma série temporal $x_t; t = 0, \pm 1, \pm 2, \dots$ é um **modelo autorregressivo de média móvel** $ARMA(p, q)$ se é estacionária e*

$$x_t = \phi_1 x_{t-1} + \dots + \phi_p x_{t-p} + \omega_t + \theta_1 \omega_{t-1} + \dots + \theta_q \omega_{t-q}, \quad (2.28)$$

onde x_t é estacionário, $\phi_p \neq 0$, $\theta_q \neq 0$, e $\sigma_\omega^2 > 0$. Os parâmetros p e q são chamados de ordem de autorregressão e de média móvel, respectivamente (SHUMWAY; STOFFER, 2017).

A partir das definições 16 e 18, pode-se representar a Equação 2.28 como

$$\phi(B)x_t = \theta(B)\omega_t. \quad (2.29)$$

2.4 PROCESSOS NÃO-ESTACIONÁRIOS

2.4.1 PROCESSOS AUTORREGRESSIVOS INTEGRADOS DE MÉDIA MÓVEL (ARIMA)

Um **modelo autorregressivo integrado de média móvel** (*integrated ARMA*) abreviado na forma *ARIMA* é um modelo mais abrangente da classe de modelos ARMA para incluir diferenciação, segundo Definição 14, com o objetivo de corrigir a não-estacionariedade em média da série temporal original.

Definição 20. *Um processo x_t é dito ser $ARIMA(p, d, q)$ se (SHUMWAY; STOFFER, 2017):*

$$\nabla^d x_t = (1 - B)^d x_t$$

é um processo $ARMA(p, q)$. De forma geral, escreve-se o modelo como

$$\phi(B)(1 - B)^d x_t = \theta(B)\omega_t \quad (2.30)$$

3 REQUISITOS DE PROJETO

O presente capítulo tem por objetivo apresentar os requisitos do projeto a ser desenvolvido, ou seja, as premissas básicas que deverão ser seguidas para o desenvolvimento da plataforma.

3.1 ESTRUTURA DA PLATAFORMA

A plataforma a ser implementada, de forma geral, precisa atender ao seguinte processo:

- 1 Escolha de uma das funções ou algoritmos disponíveis;
- 2 Importação dos dados e definição dos parâmetros da função;
- 3 Processamento e análise de resultados.

No primeiro passo, será definido um sumário para seleção da função desejada pelo usuário, além de documentação referenciando e descrevendo os parâmetros e resultados para a função selecionada.

No segundo passo, será construído um formulário para a entrada manual dos parâmetros pelo usuário, com opções padrão já definidas. Junto à definição dos parâmetros, o usuário precisa carregar os arquivos com os dados referentes à série temporal a ser analisada, em formato de arquivo CSV.

Por fim, no terceiro passo, será exibido um painel de resultados e métricas de avaliação, segundo as funções implementadas.

Com base nos requisitos acima apresentando, a Figura 3.1 apresenta um esquema da tela inicial da plataforma a ser desenvolvida.

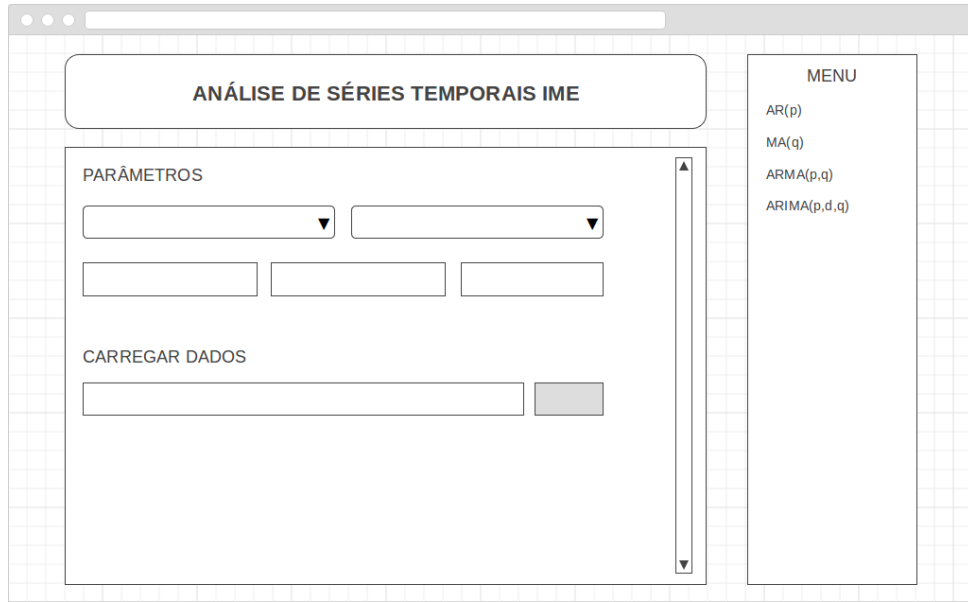


FIG. 3.1: Esquema Tela Inicial Plataforma

A plataforma será construída empregando a linguagem de programação *Python*.

3.2 ALGORITMOS

Conforme exposto no Capítulo 2, serão disponibilizados na plataforma algoritmos de análise exploratória, análise preditiva e de filtragem para Séries Temporais, sendo requisitos mínimos:

- $ACF(k)$;
- $PACF(k)$;
- $ARIMA(p, d, q)$;

Os algoritmos acima expostos precisam estar embarcados na plataforma, porém não é requisito a implementação autoral destes, devido às diferentes otimizações e avanços já consolidados em soluções vigentes.

Portanto, será utilizado inicialmente como referência o pacote de funções *statsmodels*, desenvolvido em *Python* (PERKTOLD et al., 2017).

3.3 FORMATOS DE ENTRADA DE DADOS

Como citado na Seção 3.1, o único formato de arquivo aceito pela plataforma é o CSV (extensão *".csv"*).

Não há restrição quanto ao separador empregado no arquivos, sendo esse um parâmetro fornecido pelo usuário. Pressupõe-se que o arquivo deverá conter ao menos duas colunas de dados, uma referente ao informação de tempo (datas, meses, anos, etc) e outra com o dado amostrado ao longo do tempo. Não há, porém, restrição quanto à ordenação de tais colunas.

Não há restrição de formato para a coluna de dados, desde que a mesma seja povoada com valores numéricos. Quanto à coluna de tempo são aceitos somente os formatos:

- "yyyy-mm-dd"
- "dd-mm-yyyy"
- "yy-mm-dd"
- "dd-mm-yy"

Onde "y"denota dígito referente ao ano, "m"referente ao mês, e "d"referente ao dia. Em todos os casos apresentados o separador "-"pode ser substituído pelo separador "/", sem prejuízo na leitura do formato. Em caso de dados de tempo em formatos distintos dos acima estabelecidos, o método de leitura executa um tratamento de exceção para execução dos algoritmos, mas não é garantida confiabilidade no processo.

4 METODOLOGIA

4.1 FERRAMENTAS EMPREGADAS

Para o desenvolvimento da plataforma foram definidos dois grupos de ferramentas a serem empregadas: ferramentas destinadas ao desenvolvimento do código e ferramentas destinadas ao gerenciamento e versionamento do projeto.

4.1.1 DESENVOLVIMENTO DO CÓDIGO

No que diz respeito às ferramentas relacionadas ao desenvolvimento do código, conforme abordado no Capítulo 3 a linguagem de programação escolhida para o desenvolvimento do projeto foi *Python* 3. Optou-se por empregar somente *Python* na construção da Plataforma, tanto na construção do sistema *Web* como na codificação dos algoritmos. A opção por utilizar-se *Python* também na construção do sistema *Web* foi resultante do conhecimento prévio da linguagem por parte da equipe e da maior facilidade de interfaceamento com os métodos a serem implementados ao manter-se uma única linguagem.

Para a construção da plataforma e desenvolvimento dos algoritmos foram empregadas bibliotecas da linguagem Python, conforme Apêndice 7.1.

A principal biblioteca empregada na construção da plataforma *Web* foi empregado o *microframework Flask* (RONACHER, 2018) conciliado com *WTForms* (TEAM, 2010) na construção dos formulários para os algoritmos e *jinja* (RONACHER, 2014) na integração entre *Python* e html para construção dos templates. Na construção do layout foi empregado o conjunto de ferramentas do *Bootstrap* (BOOTSTRAP, 2016).

Para os algoritmos destacam-se as bibliotecas *pandas* (PANDAS, 2018), empregada principalmente na leitura dos dados, e *matplotlib* (HUNTER et al., 2018) e *statsmodels* (PERKTOLD et al., 2017), empregadas na construção dos modelos e plotagem dos resultados.

4.1.2 GERENCIAMENTO DO PROJETO

Para o gerenciamento do desenvolvimento do projeto optou-se pelo uso do *GitHub* (GITHUB, 2018), uma plataforma de desenvolvimento que permite a construção de projetos de software de forma colaborativa mas gerenciando o versionamento e a integração

entre as diversas partes. Foi criado um repositório no *GitHub* onde a plataforma está sendo desenvolvida e será armazenado ao final do Projeto. A Figura 4.1 demonstra a estrutura do repositório empregado no *GitHub* para o gerenciamento do desenvolvimento.

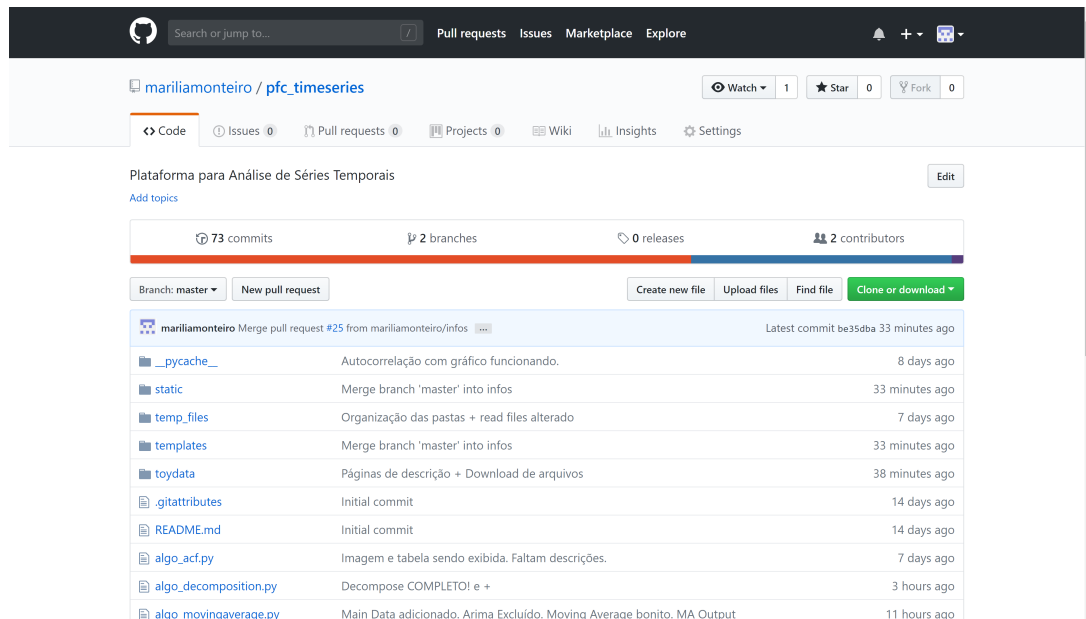


FIG. 4.1: Repositório do Projeto no GitHub

É importante enfatizar que o *GitHub* não funciona como servidor da plataforma *Web*, mas somente como repositório para os códigos, permitindo que qualquer usuário possa acessar e contribuir com a plataforma.

Além do repositório online são disponibilizadas estruturas para *Desktop* integradas com aquele que permitem o interfaceamento do desenvolvimento de forma ágil.

Paralelamente ao *GitHub*, para versionamento da plataforma foi empregada a plataforma *Zenodo* (PROJECT, 2013), usada para a disponibilização de projetos de código no meio científico e que permite a criação de versões permanentes, enquanto o *GitHub* está em constante mudança por ser uma plataforma para desenvolvimento e colaboração.

4.2 PÁGINAS ESTÁTICAS

O objetivo na criação de páginas estáticas é garantir que a plataforma seja de fácil uso e que possua interface amigável ao usuário, conforme estabelecido na Seção 1.3. As páginas estáticas aqui citadas são aquelas que contêm textos e imagens mas não realizam operações resultantes de interações com os usuários. Foram assim, construídas as seguintes páginas estáticas:

- **Página inicial:** Contém a apresentação inicial da plataforma bem como informações gerais sobre a mesma. Oferece caminhos para acessar diretamente as páginas "Algoritmos" e "Documentação". Quando acessado de forma direta (sem endereço pré-estabelecido), o site é direcionado para essa página.
- **Documentação:** Contém informações mais detalhadas do que a "Página Inicial" sobre o projeto e referências relacionadas ao mesmo.
- **Algoritmos:** Apresenta os métodos disponíveis para exploração de dados temporais. Os algoritmos são apresentados em cartões, contendo categoria, título e descrição simplificada.
- **Formatos de Arquivos:** Descreve os formatos e a estrutura de arquivos aceitos para *upload* nos algoritmos disponíveis na plataforma.
- **Exemplos:** Disponibiliza conjuntos de dados e parâmetros associados aos mesmos que podem ser utilizados como exemplo para o teste dos algoritmos disponíveis na plataforma.

4.3 ESTRUTURA DOS ALGORITMOS

Enquadram-se como páginas dinâmicas — aquelas que permitem interação com o usuário — todas as páginas dos algoritmos, usadas para o processamento de dados. O objetivo da presente seção é, portanto, apresentar o fluxo de funcionamento empregado na construção das páginas de algoritmos adicionadas à plataforma. A apresentação aqui descrita é feita de modo genérico, sem que sejam abordadas, portanto, as funções específicas de cada um dos modelos implementados; trata-se, assim, de uma visão geral do processo de implementação e funcionamento das páginas de algoritmos.

Para cada algoritmo criado três elementos visuais devem ser adicionados à plataforma: cartão na página de algoritmos contendo informações básicas do modelo e que direciona para página do algoritmo propriamente dito (Figura 4.2); página para envio de parâmetros e dados (Figura 4.3); e página para exibição dos resultados após processamento dos algoritmos (Figura 4.4).



FIG. 4.2: Página de Apresentação dos Algoritmos

Parâmetros

Lags*

 Número de Lags (diferenças) a serem consideradas

Upload de Arquivo (Saiba mais)
 meansealevel.csv

Separador* ☒ Cabeçalho*

Coluna de datas Coluna principal

Descrição

Função de autocorrelação total da série principal dos dados utilizados. Retorna gráfico e tabela apresentando os valores de autocorrelação para cada lag (atraso nos dados, em quantidade de linhas), começando em 0 (correlação 1).
 Parâmetros de entrada:

- Lags: define a quantidade de autocorrelações que serão exibidas, começando com a correlação com a série original (lag zero) e terminando com o atraso no valor definido no formulário.

FIG. 4.3: Exemplo de formulário para envio de informações

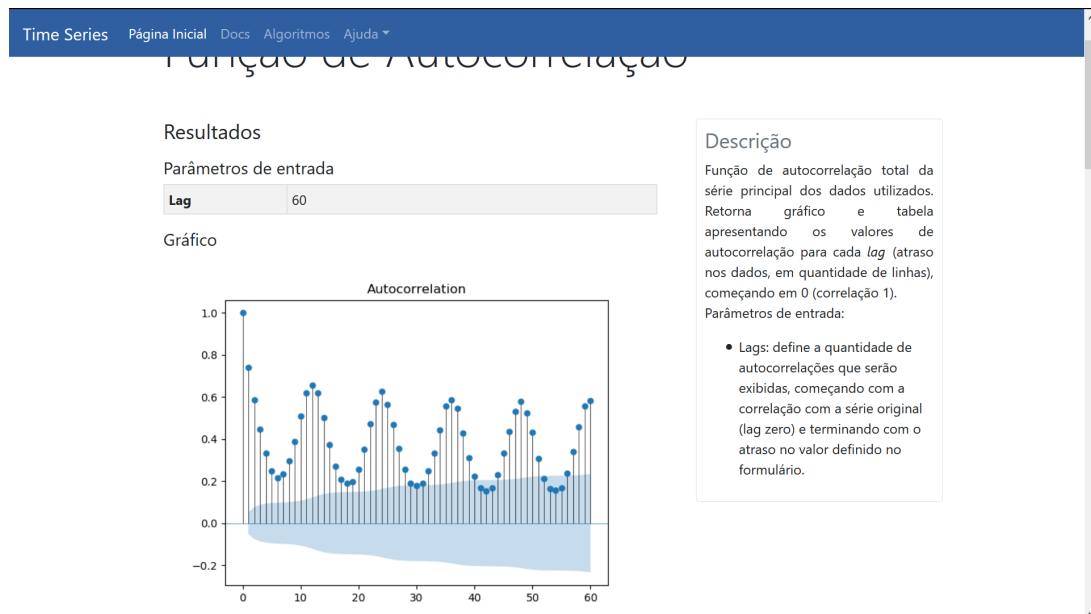


FIG. 4.4: Exemplo de página para exibição de resultados

A adição de novos cartões é automática, desde que sejam criados os textos nas pastas `pfc_timeseries\static\texts_long` e `pfc_timeseries\static\texts_short` conforme apresentado na Subseção 4.4.2 (mais detalhes sobre a estrutura das pastas serão apresentados na Seção 4.4). A adição automática é realizada através do uso de funções definidas no arquivo `variables.py` associadas com a criação de um template dinâmico em `\templates\algorithms.html` empregando *jinja* (RONACHER, 2014).

Para cada algoritmo adicionado, deve ser criada em `app.py` uma nova função associada a um novo endereço conforme estrutura apresentada no Exemplo de Código 4.1.

Código 4.1: Adição de nova página de algoritmo

```

1  @app.route('/algorithms/acf', methods= ['GET', 'POST'])
2  def algorithms_acf():
3      text = desc_list['01 acf']
4      form = FormACF()
5
6      if form.validate_on_submit():
7          (...)
8          return render_template('algorithms_nome_output.html',
9                                 title='Titulo', ...)
10     else:

```

```

11         file_url = None
12
13     return render_template('algorithms_nome.html', title='Titulo
        ', text=text, form= form, file_url=file_url)

```

Para novos algoritmos, especificamente, a função deverá ser capaz de:

- **Importar textos de descrição:** trata-se da importação do texto contendo a descrição do algoritmo acessando-o na lista de texto de descrição através do nome do arquivo livre de sua extensão. O objeto do texto deverá ser passado como variável na renderização do *template* da página de envio de informações do algoritmo e exibição de resultados, uma vez que a descrição do algoritmo é apresentada em ambas as páginas.
- **Criar objeto da classe do formulário:** criação de objeto da classe do formulário adicionado a `forms.py`. O objeto deve ser passada como variável na renderização do *template* da página do algoritmo para que os campos do formulário sejam criados e funcionem de modo adequado.
- **Exibir página para envio de informações:** a página contendo o formulário com os campos referentes aos parâmetros do modelo e ao envio de arquivo deve ser exibida inicialmente, para que possa ser preenchida pelo usuário. Assim deve-se renderizar o html da página do formulário, sendo ela construída utilizando *jinja* (RONACHER, 2014) e *WTForms* (TEAM, 2010). Deve ser possível o envio (submissão) dos parâmetros e consequente validação. O resultado esperado é exemplificado na Figura 4.3
- **Alterar o nome do arquivo enviado pelo usuário:** após o envio dos parâmetros e dados, o arquivo de texto CSV recebido tem seu nome alterado com o objetivo de manter o sigilo dos dados do usuário e evitar problemas com títulos repetidos na pasta de arquivos temporários. Após a alteração do nome do arquivo, o mesmo é armazenado em `\temp_files`. O processo é exemplificado no trecho de Código 4.2.

Código 4.2: Alteração de nome de arquivos

```

1 filename = secrets.token_hex(8) + '.csv'
2 form.dados.data.save(os.path.join(app.config['UPLOAD_FOLDER']
    ], filename))
3 file_url = files.url(filename)

```


- **Armazenar as informações passadas através formulário:** as informações advindas do formulário devem ser tratadas e armazenadas em variáveis de modo a serem utilizadas no processamento dos dados.
- **Ler arquivo enviado pelo usuário:** a leitura do arquivo é realizada em todos os algoritmos empregando a função de leitura de arquivo definida em `read_file.py`. Os formatos de arquivo aceitos e suas características estruturais são apresentados na Seção 3.3.
- **Realizar o processamento dos dados:** com base nos parâmetros armazenados a partir do formulário e nos dados interpretados com base no arquivo recebido, é realizado processamento dos algoritmos empregando funções importadas dos arquivos do tipo `algo_<nome-do-algoritmo>.py`. As saídas dos algoritmos poderão ser conjuntos de dados ou imagens (gráficos). As imagens geradas são salvas de maneira temporária na pasta `\static\images`.
- **Gerar página de resultados:** a criação da página de resultados é feita através da renderização de um *template* criado para tal, conforme abordado em 4.4.2. O *template* é construído empregando *jinja* (RONACHER, 2014), e recebe por variável os conjuntos de dados de saída ou as referências das imagens.

São, ainda, atividades paralelas ao fluxo de execução de um algoritmo os processos de exclusão do arquivo importado pelo usuário e de exclusão das imagens geradas no processamento dos algoritmos. Os arquivos enviados pelos usuários são excluídos sempre que a leitura do arquivo é feita, ou seja, sempre que a função `read_csv` disponível em `read_files.py` é empregada. As imagens geradas não podem ser excluídas automaticamente, como é o caso dos arquivos ".csv" que podem ser excluídos sempre que os dados, ou seja, a informação de interesse, são extraídos deles e armazenados em outro objeto; isso se deve a necessidade de referência do "caminho" das imagens (localização no repositório) para que sejam exibidas nas páginas de resultados; assim, a rotina `delete_images` disponível em `delete_images.py` é implementada separadamente de modo a deletar da pasta que contém as imagens aquelas mais antigas que um intervalo de tempo definido como variável global em `app.py`. O intervalo de tempo empregado atualmente é 1 (uma) hora. A função é chamada sempre que a página "Página Inicial" ou "Algoritmos" é acessada (ver Seção 4.2 para mais detalhes sobre as páginas citadas).

4.4 ESTRUTURA DA PLATAFORMA

Com base na estrutura estabelecida pelo *microframework Flask*, empregado na construção da Plataforma *Web*, a plataforma é organizada conforme segue:

- `\pfc_timeseries`: repositório principal contendo arquivos e subpastas usados na construção da plataforma.
 - `app.py`: arquivo Python principal, responsável pela estruturação de toda a plataforma, suas páginas e processos.
 - `forms.py`: arquivo contendo classes herdeiras da classe *FlaskForm* empregadas na construção dos formulários para os algoritmos.
 - `read_file.py`: código contendo a função de leitura dos arquivos importados pelo usuário, única para toda a plataforma.
 - `variables.py`: código contendo funções específicas empregadas na criação de novas páginas de algoritmos.
 - `delete_images.py`: código contendo função empregada na limpeza de pastas contendo imagens geradas de forma temporária.
 - `algo_<nome>.py`: códigos com funções criadas para cada um dos algoritmos e métodos implementados na plataforma.
 - `README.md`: arquivo contendo instruções gerais sobre a estrutura da plataforma.
 - `..\static`
 - * `..\images`: pasta de arquivos temporários contendo figuras e gráficos gerados no processamento dos dados. A pasta é esvaziada periodicamente. Os arquivos adicionados à pasta são criados automaticamente no processamento da maioria dos algoritmos e possuem nomes gerados aleatoriamente.
 - * `..\texts_long`: pasta contendo arquivos do tipo texto (".txt") com as descrições de cada um dos algoritmos empregados.
 - * `..\texts_short`: pasta contendo arquivos do tipo texto (".txt") com os dados principais necessários à criação de um algoritmo na plataforma.
 - `..\temp_files`: pasta que armazena temporariamente os arquivos importados pelo usuário. Os arquivos tem o nome alterado na entrada para um conjunto de caracteres aleatório e são excluídos assim que os dados são extraídos dos mesmo.

- `..\templates`: pasta contendo arquivos ".html" para todas as páginas do site. É também empregado *Jinja2* (RONACHER, 2014) na construção dos *templates*.
- `..\toydata`: pasta contendo arquivos no formato CSV utilizados como exemplos de conjuntos de dados na plataforma.

A plataforma de forma geral foi construída de modo a permitir a fácil edição e inserção de novos algoritmos e páginas. Nas subseções a seguir serão apresentadas algumas das principais estruturas empregadas de modo a viabilizar a continuidade e colaboratividade do projeto.

4.4.1 ADIÇÃO DE NOVAS PÁGINAS

Para a criação de uma nova página no site devem ser executados os seguintes passos:

- 1 **Criação de *Template* no formato "html"**: deve ser adicionado um *template* no formato "html" na pasta `templates`. De modo a garantir-se a uniformidade de design da plataforma deve-se estender o *template* padrão do site usando a estrutura "*extends*" e "*block*" do *jinja* conforme apresentado no extrato de Código 4.3. Pode-se empregar variáveis adicionais conforme documentação do *jinja* (RONACHER, 2014).

Código 4.3: Extensão de Template

```

1  {% extends "layout.html" %}
2
3  {% block content %}
4  <div class="container" style="margin-bottom:50px; margin-top
    :40px;">
5    <h1 class="display-4">Titulo da pagina</h1>
6  </div>
7  <!-- Conteudo da pagina -->
8  {% endblock content %}
```

- 2 **Adição de Página**: adiciona-se a página com endereço e função executada quando a página é executada que deve incluir a exibição do template html. O extrato de Código 4.4 apresenta exemplo de adição de página.

Código 4.4: Adição de Página de Documentação

```
1 @app.route('/docs')
2 def about():
3     return render_template('docs.html', title='Docs')
```

4.4.2 ADIÇÃO DE NOVOS ALGORITMOS

Para adição de novos algoritmos devem ser executadas as seguintes etapas:

- 1 **Adição de nova classe de formulário em forms.py:** deve ser criada uma nova classe específica para o formulário que se deseja construir contendo a estrutura dos campos para o formulário do novo algoritmo. Existem campos fixos a todas as classes conforme comentário em classes já criadas, destinados a ao *upload* de arquivos, que devem ser repetidos em novas classes. O extrato de Código 4.5 apresenta a estrutura de uma possível classe, com novos campos no início ("*window*") e os campos obrigatórios em seguida, conforme comentário.

Código 4.5: Adição de classe para novo formulário

```
1 class FormMA(FlaskForm):
2     window = IntegerField('Janela', default=3, validators=[
3         InputRequired()])
4     # CAMPOS FIXOS PARA TODOS OS FORMULARIOS (ARQUIVO)
5     dados = FileField('Dados', validators=[FileAllowed(['csv',
6         'Somente arquivos .csv']), FileRequired()])
7     sep = StringField('Separador', default=',', validators=[
8         Length(min=1, max=2), InputRequired()])
9     header = BooleanField('Cabecalho', default=True)
10    datec = IntegerField('Coluna de datas', default=1)
11    datac = IntegerField('Coluna principal', default=2)
12    submit = SubmitField('Processar')
```

- 2 **Textos de Descrição:** devem ser criados novos arquivos do tipo ".txt" nas pastas `pfc_timeseries\static\texts_long` e `pfc_timeseries\static\texts_short` com o mesmo nome em ambas as pastas. O arquivo na pasta `pfc_`

`timeseries\static\texts_long` deverá ser um texto corrido, podendo ser no formato html, contendo a descrição do algoritmo, sendo exibido na barra lateral de descrição da página do método. O arquivo na pasta `pfc_timeseries\static\texts_short` deverá respeitar a estrutura apresentada no Código 4.6.

Código 4.6: Adição de classe para novo formulário

```
1 page: <caminho da pagina (apos "\algorithms\")>
2 title: <Titulo para a pagina>
3 subtitle: <Tipo de Analise>
4 desc: <Breve descricao do metodo>
```

Com ambos os arquivos criados o algoritmo terá uma opção na página de algoritmos automaticamente adicionada.

- 3 **Construção do html:** deve ser construída a página html na pasta `\pfc_timeseries\templates`. Para tanto deve-se estender o conteúdo da página `\pfc_timeseries\templates\layout_algorithms.html` seguindo a estrutura apresentada no Código 4.3, mas preenchendo o "block" chamado "formulario". Para a construção do html são utilizadas estruturas de formulário do WTForms (TEAM, 2010).
- 4 **Implementação do Método:** o método deve ser implementado em arquivo do tipo *Python* a parte, sendo construído dentro de funções, as quais serão importadas em `app.py` para utilização na página do método.
- 5 **Criação da página em `app.py`:** deve-se seguir a estrutura apresentada na Subseção 4.4.1 com atenção na definição das funções, para que seja definido o formulário com os parâmetros necessários. É importante atentar às variáveis necessárias à criação da página de *upload* de dados e de apresentação dos resultados.
- 6 **Construção do html dos resultados de saída:** deve ser construída a página html na pasta `\pfc_timeseries\templates`. Para tanto deve-se estender o conteúdo da página `\pfc_timeseries\templates\layout_algorithms_output.html` seguindo a estrutura apresentada no Código 4.3, mas preenchendo o "block" chamado "formulario". Para a construção do html são utilizadas estruturas de formulário do WTForms (TEAM, 2010).

As etapas acima apresentadas são interdependentes, ou seja, para o funcionamento do novo algoritmo adicionado é necessário que todas as partes tenham sido corretamente im-

plementadas. Maiores detalhes sobre o fluxo de funcionamento das páginas dos algoritmos são apresentados na Seção 4.3.

5 PLATAFORMA *WEB* (VERSÃO 1.0)

O capítulo visa apresentar a primeira versão da plataforma *Web* desenvolvida. A versão aqui citada da plataforma encontra-se disponível conforme (MONTEIRO; MARCHIORE, 2018).

Para apresentar a versão atual da plataforma serão apresentadas as principais telas da mesma, bem como os algoritmos já implementados e os resultados de alguns testes utilizando os dados disponibilizados na página de "Exemplos" empregando tais algoritmos.

5.1 PÁGINA INICIAL

A **Página Inicial**, apresentada na Figura 5.1 é a página para onde o usuário é direcionado assim que acessa o servidor da plataforma. Atendendo ao descrito na Seção 4.2, ela trata-se de uma página estática, que pode direcionar para a página de Documentação, através do botão "Saiba Mais", e também para a página de "Algoritmos" através do botão "Conheça os algoritmos".

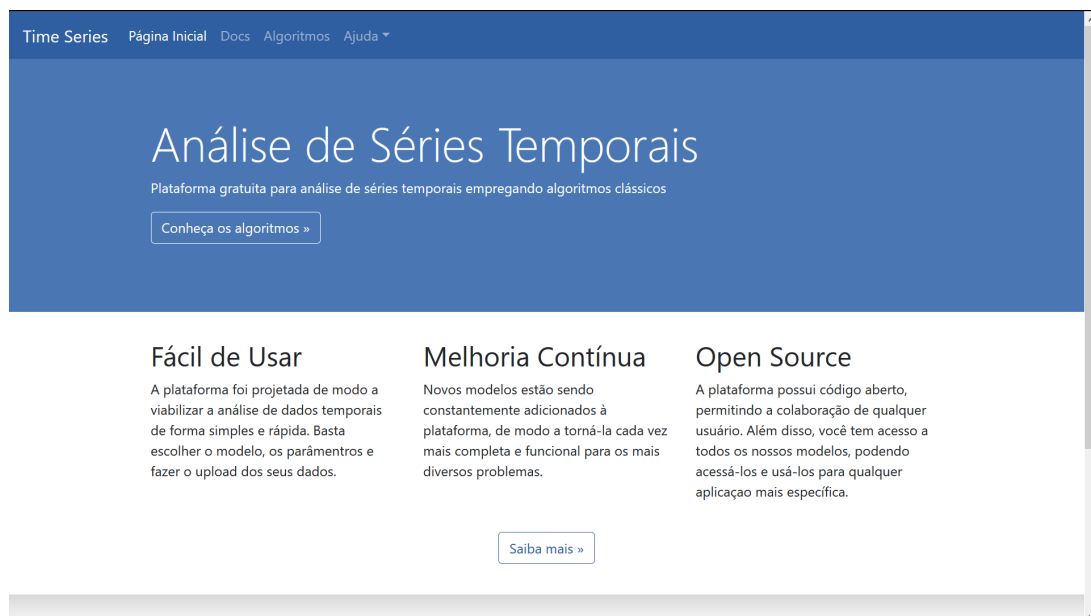


FIG. 5.1: Página Inicial - Versão 1.0

Além dos caminhos para acessar outras páginas estabelecidos nos citados botões, em todas as páginas da plataforma está disponível a barra superior que permite navegar para todas as páginas criadas de forma rápida.

5.2 DOCUMENTAÇÃO

A página de **Documentação**, apresentada na Figura 5.2, conforme citado na Seção 4.2, apresenta uma maior descrição sobre a plataforma e sobre o Projeto de desenvolvimento da mesma; cita as principais bibliotecas empregadas na construção do sistema, referenciando ainda o repositório no GitHub, o que atende às premissas estabelecidas na Seção 1.3, onde estabelece-se que a plataforma deverá ser *Open-Source*.



FIG. 5.2: Documentação - Versão 1.0

5.3 FORMATO DE ARQUIVOS

Sob a opção do menu **Ajuda**, encontram-se os links para as páginas Formato de Arquivos e Exemplos. Na página de **Formato de Arquivos**, apresentada na Figura 5.3, encontram-se as definições de formatos de entrada e organização dos dados dos arquivos enviados para processamento pelos usuários. As definições seguem o descrito na Seção 3.3.



FIG. 5.3: Formato de Arquivos - Versão 1.0

5.4 EXEMPLOS

Ainda sob a aba Ajuda, na página **Exemplos**, Figura 5.4, encontram-se disponibilizadas algumas bases de dados em formato *CSV*, com a explicação sobre origem e significado dos dados, configurações ideais para utilizar o arquivo e as referências dos órgãos responsáveis pela construção e disponibilização original dos dados.



FIG. 5.4: Exemplos de dados - Versão 1.0

É possível realizar o *download* dos arquivos direto do site, sem que o usuário precise

ser redirecionado para outros endereços *Web*, como apresentado na Figura 5.5.

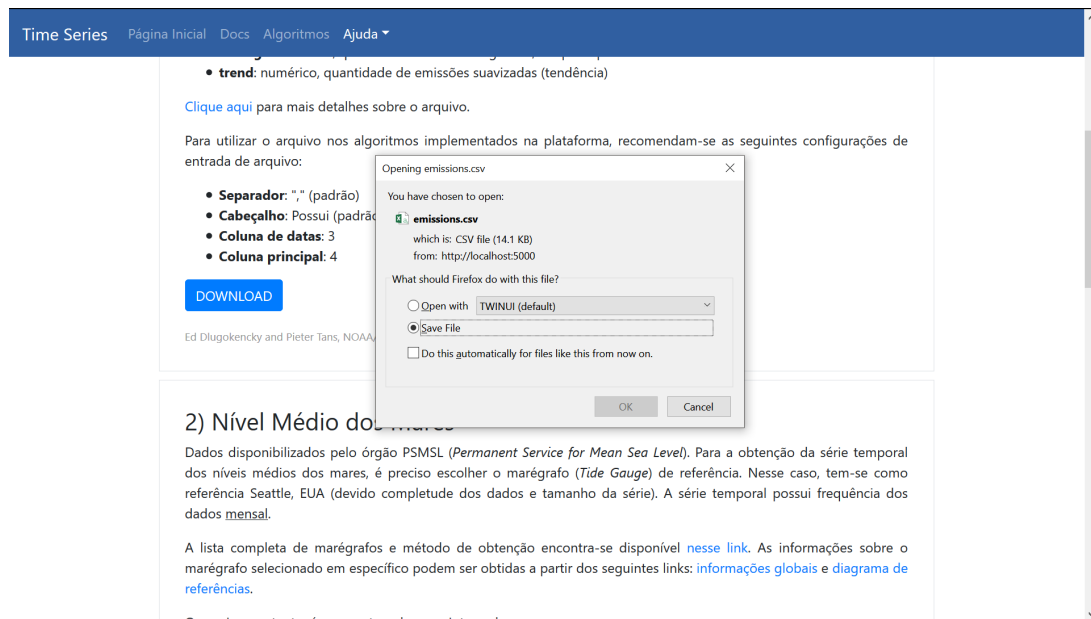


FIG. 5.5: Exemplos, download dos arquivos

5.5 ALGORITMOS

A página de Algoritmos apresenta todos os algoritmos para processamento de dados de séries temporais disponíveis no sistema. Cada algoritmo é apresentado em um cartão, conforme descrito na Seção 4.2. A apresentação atual da página pode ser visualizada na Figura 5.6.

Para fins de classificação dos cartões, os métodos disponíveis podem pertencer às seguintes classes:

- **Análise Exploratória:** refere-se às etapas e processos iniciais de análise da Série Temporal, onde o objetivo é obter estatísticas descritivas, procurar padrões iniciais e realizar testes — como testes de estacionariedade.
- **Filtragem:** refere-se aos processos de alteração dos dados originais da Série Temporal de modo a realçar determinado comportamento, remover características indesejadas ou obter padrões. São exemplos filtros lineares, operadores de transformações de dados e decomposição de Série.
- **Modelagem Preditiva:** refere-se às etapas de modelagem do processo estocástico que melhor caracteriza os dados da Série Temporal, de modo a realizar estimativas futuras sobre os valores esperados para a Série.



FIG. 5.6: Algoritmos - Versão 1.0

Exemplos de algoritmos disponíveis em Python para cada classe anteriormente descrita podem ser acessados no *statsmodels* (PERKTOLD et al., 2017).

Na corrente versão são disponibilizados os seguintes métodos (algoritmos):

- Função de Autocorrelação;
- Função de Autocorrelação Parcial;
- Decomposição de Série; e
- Filtro de Média Móvel.

As páginas de entrada (definição dos parâmetros) e saída (resultados) da cada algoritmo serão apresentadas nas subseções a seguir.

5.5.1 FUNÇÃO DE AUTOCORRELAÇÃO

Para a Função de Autocorrelação, o único parâmetro de entrada consiste no número de *Lags*, ou seja, na quantidade de autocorrelações que serão exibidas no gráfico, variando de 0 até o número passado. Por exemplo, caso seja informado o número 20 de *Lags*, o gráfico resultante apresentará um total de vinte autocorrelações, variando o atraso desde nenhum intervalo de dados (resultando em 1), até dezenove intervalos de dados. Os cálculos estão de acordo com o apresentado na Equação 2.3.

A Figura 5.7 apresenta a entrada para a utilização do algoritmo em questão; a Figura 5.8 apresenta as saídas da função. Além do gráfico principal referente à operação utilizada,

os algoritmos também exibem os dados resultantes em formato de tabela, para que possam ser extraídos e utilizados pelo usuário, caso seja de interesse.

Time Series
Página Inicial
Docs
Algoritmos
Ajuda

Função de Autocorrelação

Parâmetros

Lags*

Número de Lags (diferenças) a serem consideradas

Upload de Arquivo (Saiba mais)

meansealevel.csv

Separador* ☒ Cabeçalho*

Coluna de datas Coluna principal

Descrição

Função de autocorrelação total da série principal dos dados utilizados. Retorna gráfico e tabela apresentando os valores de autocorrelação para cada *lag* (atraso nos dados, em quantidade de linhas), começando em 0 (correlação 1).

Parâmetros de entrada:

- Lags: define a quantidade de autocorrelações que serão exibidas, começando com a correlação com a série original (lag zero) e terminando com o atraso no valor definido no formulário.

FIG. 5.7: Função de Autocorrelação - Versão 1.0

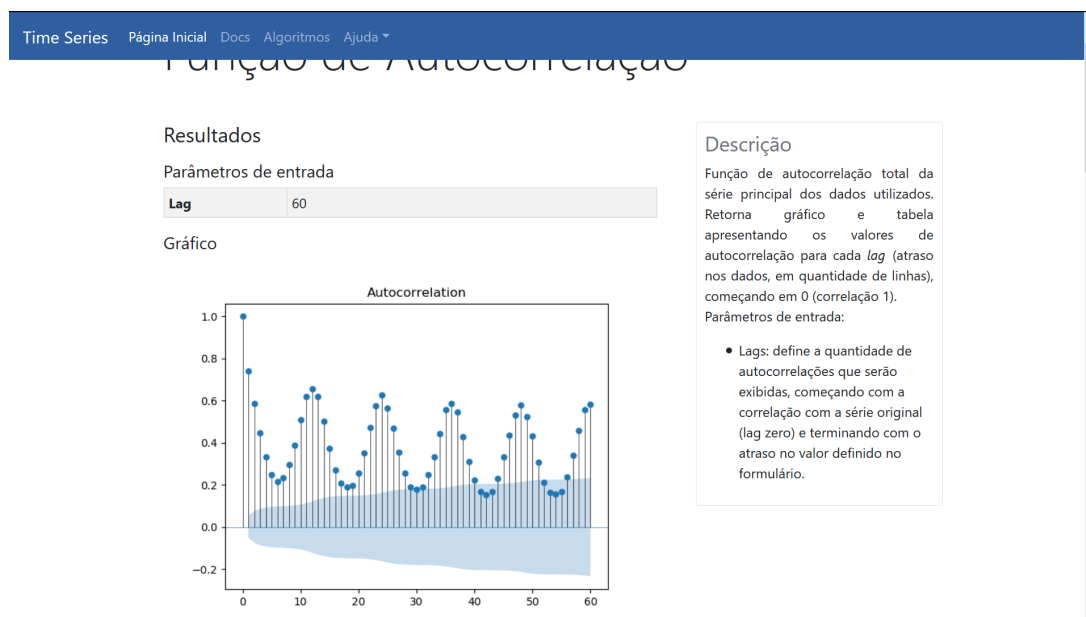


FIG. 5.8: Resultados da Função de Autocorrelação - Versão 1.0

5.5.2 FUNÇÃO DE AUTOCORRELAÇÃO PARCIAL

Os parâmetros de entrada e resultados exibidos para a Função de Autocorrelação Parcial assemelham-se aos da Função de Autocorrelação. As diferenças teóricas são explicadas

na Definição 4. As Figuras 5.9 e 5.10 apresentam os parâmetros de entrada e resultados da função, respectivamente.

Time Series

Página Inicial

Docs

Algoritmos

Ajuda

Função de Autocorrelação Parcial

Parâmetros

Lags

60

Número de Lags (diferenças) a serem consideradas

Upload de Arquivo (Saiba mais)

Browse...

meansealevel.csv

Separador

,

☒ Cabeçalho

Coluna de datas

1

Coluna principal

2

Processar

Descrição

Função de autocorrelação parcial da série principal dos dados utilizados. Retorna gráfico e tabela apresentando os valores de autocorrelação parcial para cada *lag* (atraso nos dados, em quantidade de linhas), começando em 0 (correlação 1). Diferencia-se com relação à autocorrelação total por remover qualquer dependência linear que possa existir com dados de *lags* inferiores.

Parâmetros de entrada:

- Lags: define a quantidade de autocorrelações que serão exibidas, começando com a correlação com a série original (lag zero) e terminando com o

FIG. 5.9: Função de Autocorrelação Parcial - Versão 1.0



FIG. 5.10: Resultados da Função de Autocorrelação Parcial - Versão 1.0

5.5.3 DECOMPOSIÇÃO DE SÉRIES

O algoritmo de Decomposição de Séries implementa o Método de Decomposição Clássica de Séries Temporais, conforme apresentado na Seção 2.2.2. Os parâmetros de entrada do algoritmo na página são determinados pelo usuário seguindo o exposto na Figura 5.11.

Time Series
Página Inicial
Docs
Algoritmos
Ajuda

Decomposição de Séries

Parâmetros

Modelo

Aditivo

Tipo de modelo a ser utilizado na decomposição

Frequência dos Dados

Mensal

Frequência de coleta dos dados

Sazonalidade dos Dados

Anual

Sazonalidade esperada ou conhecida dos dados

Referência Média Móvel

À direita

Centralização da referência para média móvel

Outra Frequência

0

Input manual da frequência. Para preencher assinala "Outros" ao lado

Outra Sazonalidade

0

Input manual da sazonalidade. Para preencher assinala "Outros" ao lado.

Upload de Arquivo

(Saiba mais)

Descrição

Decomposição da série temporal em componentes representantes de diferentes padrões, sendo elas: tendência (*Trend*), sazonalidade (*Seasonal*) e aleatoriedade ou resíduo (*Residual*). Retorna gráficos de cada uma das componentes.

Parâmetros de entrada:

- Modelo:** define o modelo a ser empregado para a decomposição da série, podendo ser aditivo ou multiplicativo. O modelo aditivo é linear, admitindo que as componentes resultantes (tendência, sazonalidade e resíduo) somadas resultam no dado observado. Já o modelo

FIG. 5.11: Decomposição de Séries - Versão 1.0

Para a obtenção das Séries Temporais resultantes, é preciso que se conheça *à priori* a janela de repetição do efeito de sazonalidade. Na plataforma, isso é feito a partir da razão entre os dados obtidos das opções "Frequência dos Dados"(que representa o intervalo de tempo entre registros/amostras na série de dados) e "Sazonalidade dos Dados"(que representa o intervalo de tempo em que há sazonalidade percebida, ou seja, intervalo de tempo em que se espera que os dados tenham seu comportamento repetido – tempo do ciclo).

O exemplo dos resultados observados pelo usuário está apresentado na Figura 5.12.



Time Series
Página Inicial
Docs
Algoritmos
Ajuda

Média Móvel

Parâmetros

Janela

90

Tamanho da janela (intervalo) para média móvel

Upload de Arquivo

(Saiba mais)

Browse...
gps.csv

Separador

,

☒ Cabeçalho

Coluna de datas

1

Coluna principal

4

Processar

Descrição

Filtragem dos dados empregando média móvel. Retorna o gráfico dos dados filtrados e os valores ponto a ponto da série após a filtragem em tabela.

Parâmetros de entrada:

- Janela: define a quantidade de elementos que comporão a janela na operação de média, ou seja, a largura da janela. Quanto maior a janela, mais suavizada será a série de saída.

FIG. 5.13: Função de Média Móvel - Versão 1.0



FIG. 5.14: Resultados da Função de Média Móvel - Versão 1.0

6 REFERÊNCIAS BIBLIOGRÁFICAS

- BOOTSTRAP. Bootstrap. Disponível em: <<https://getbootstrap.com/>>. Acesso em: 30 de julho de 2018.
- BROCKLEBANK, J. C.; DICKEY, D. A. **SAS® for Forecasting Time Series**. 2. ed. Cary, NC, USA: SAS Institute Inc, 2003. 418 p.
- BROCKWELL, P. J.; DAVIS, R. A. **Introduction to Time Series and Forecasting**. 2. ed. New York: Springer, 2002. 181 p.
- CORDEIRO, C.; CRISTINA, S.; GOELA, P.; DANCHENKO, S.; ICELY, J.; LAVENDER, S. ; ALICE, N. Time series analysis of remote sensing data. **Jornadas de Classificação e Análise de Dados, JOCLAD**, v. 23, p. 1–3, 2016.
- CORDEIRO, CLARA AND CRISTINA, SÓNIA AND GOELA, PRISCILA AND DANCHENKO, SERGEI AND ICELY, JOHN AND LAVENDER, SAMANTHA AND ALICE, NEWTON. Time series analysis of remote sensing data. Disponível em: <<https://goo.gl/FM3qeC>>. Acesso em: 30 de abril de 2018.
- DIDOVA, O.; GUNTER, B.; RIVA, R.; KLEES, R. ; ROESE-KOERNER, L. An approach for estimating time-variable rates from geodetic time series. **Journal of Geodesy**, v. 90, n. 11, p. 1207–1221, 2016.
- FALK, M.; MAROHN, F.; MICHEL, R.; HOFMANN, D. ; MACKE, M. **A First Course on Time Series Analysis**. 1. ed. Würzburg, Germany: University of Würzburg, 2005. 214 p.
- FERREIRA, P. G. C. **Análise de Séries Temporais em R: curso introdutório**. 1. ed. Rio de Janeiro: Elsevier - FGV IBRE, 2018. 264 p.
- GITHUB. GitHub, Built for developers. Disponível em: <<https://github.com/>>. Acesso em: 30 de julho de 2018.
- GOSSEL, W.; LAEHNE, R. Applications of time series analysis in geosciences: an overview of methods and sample applications. **Hydrology and Earth System Sciences Discussions**, v. 10, p. 12793–12827, 2013.

HUNTER, JOHN AND DALE, DARREN AND FIRING, ERIC AND DROETTBOOM, MICHAEL AND MATPLOTLIB DEVELOPMENT TEAM. Python Data Analysis Library. Disponível em: <<https://matplotlib.org/>>. Acesso em: 29 de julho de 2018.

KOSEK, WIESŁAW. IC-SG9 Application of time series analysis in geodesy - Introduction. Disponível em: <<http://users.cbk.waw.pl/~kosek/ICSG9/intro.html>>. Acesso em: 01 de maio de 2018.

LEONARD, MICHAEL AND SAMY, RENEE. Forecasting Geographic Data. Disponível em: <<https://support.sas.com/rnd/app/ets/papers/geographicforecast.pdf>>. Acesso em: 1º de maio de 2018.

MONTEIRO, MARÍLIA NASCIMENTO AND MARCHIORE, TIAGO NASCIMENTO. Primeira versão da plataforma (VC). Disponível em: <<https://doi.org/10.5281/zenodo.1323907>>. Acesso em: 30 de julho de 2018.

AL NOSEDAL. Partial Autocorrelation Function, PACF. Disponível em: <<https://mcs.utm.utoronto.ca/~nosedal/sta457/pacf.pdf>>. Acesso em: 18 de maio de 2018.

PANDAS. Python Data Analysis Library. Disponível em: <<https://pandas.pydata.org/>>. Acesso em: 29 de julho de 2018.

PERKTOLD, JOSEF AND SEABOLD, SKIPPER AND TAYLOR, JONATHAN. StatsModels: Statistics in Python. Disponível em: <<http://www.statsmodels.org/stable/index.html>>. Acesso em: 17 de maio de 2018.

OPENAIRE PROJECT. About Zenodo. Disponível em: <<http://about.zenodo.org/>>. Acesso em: 31 de julho de 2018.

R. What is R?. Disponível em: <<https://www.r-project.org/about.html>>. Acesso em: 03 de maio de 2018.

RONACHER, ARMIN. Jinja. Disponível em: <<http://jinja.pocoo.org/>>. Acesso em: 30 de julho de 2018.

RONACHER, ARMIN. Flask, web development, one drop at a time. Disponível em: <<http://flask.pocoo.org/>>. Acesso em: 29 de julho de 2018.

SHUMWAY, R. H.; STOFFER, D. S. **Time Series Analysis - Using the R Statistical Package**. 4. ed. [S.l.]: Free Dog Publishing, 2017. 181 p.

STATA. Stata© Time-Series Reference Manual Release 15. Disponível em: <<https://www.stata.com/manuals/ts.pdf>>. Acesso em: 02 de maio de 2018.

WTFORMS TEAM. WTForms Documentation. Disponível em: <<https://wtforms.readthedocs.io/en/stable/>>. Acesso em: 29 de julho de 2018.

VELSINK, H. Time series analysis of 3d coordinates using nonstochastic observations. **Journal of Applied Geodesy**, v. 10, n. 1, p. 15–16, 2016.

WESSA, PATRICK. Free Statistics and Forecasting Software. Disponível em: <<https://www.wessa.net/tsa.wasp>>. Acesso em: 03 de maio de 2018.

7 APÊNDICES

APÊNDICE 1: BIBLIOTECAS EMPREGADAS

Segue abaixo as bibliotecas da linguagem de programação *Python* empregadas na construção da plataforma. Todas as bibliotecas apresentadas são requisitos de instalação para o funcionamento da plataforma em servidor local.

- matplotlib
- statsmodels
- secrets
- os
- pandas
- flask
- flask_uploads
- werkzeug
- time
- flask_wtf
- wtforms
- numpy
- re
- codecs