



**Universidade Federal de Santa Catarina
Centro Tecnológico
Departamento de Engenharia Civil**

Ricardo Henrique Nunes

**Automação do procedimento para obtenção da orientação de pistas de pouso e
decolagem e da temperatura de referência de sítios aeroportuários**

22 de novembro de 2019

Ricardo Henrique Nunes

Automação do procedimento para obtenção da orientação de pistas de pouso e decolagem e da temperatura de referência de sítios aeroportuários

Trabalho de Conclusão de Curso apresentado pelo acadêmico Ricardo Henrique Nunes à banca examinadora do Curso de Graduação em Engenharia Civil da Universidade Federal de Santa Catarina como requisito parcial para obtenção do título de Engenheiro Civil.

Professor orientador: Alexandre Hering Coelho, Dr.

22 de novembro de 2019

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Nunes, Ricardo Henrique

Automação do procedimento para obtenção da orientação de
pistas de pouso e decolagem e da temperatura de referência
de sítios aeroportuários / Ricardo Henrique Nunes ;
orientador, Alexandre Hering Coelho, 2019.

109 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro Tecnológico,
Graduação em Engenharia Civil, Florianópolis, 2019.

Inclui referências.

1. Engenharia Civil. 2. Aeródromos. 3. Orientação da
pista de pouso e decolagem. 4. Temperatura de referência.
5. Processamento de dados. I. Coelho, Alexandre Hering.
II. Universidade Federal de Santa Catarina. Graduação em
Engenharia Civil. III. Título.

Ricardo Henrique Nunes

Automação do procedimento para obtenção da orientação de pistas de pouso e decolagem e da temperatura de referência de sítios aeroportuários

Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de Engenheiro Civil e aprovado em sua forma final pelo Departamento de Engenharia Civil da Universidade Federal de Santa Catarina.

Florianópolis, 22 de novembro de 2019

Prof^a. Luciana Rohde, Dr.^a
Coordenador do Curso de Graduação de Engenharia Civil

Banca Examinadora:

Prof. Alexandre Hering Coelho, Dr.
Orientador

Prof. Liseane Padilha Thives, Dr.
Membro

Eng. José dos Santos de Magalhães
Membro

Dedico este trabalho aos meus avós.

Agradecimentos

Agradeço aos meus pais, Roselei e Jalmir, e à minha família por todo o apoio e incentivo nos anos que antecederam e nos durante a graduação, e principalmente nesta etapa final da minha formação. Serei eternamente grato.

Ao meu irmão Jorge, que desde o início da graduação me apoiou, auxiliou e orientou. Por fazer essa caminhada parecer mais branda.

À minha namorada e amiga Priscila de Souza, que me incentivou a seguir nesta carreira. Ouviu todos meus sonhos e reclamações, sempre com muita calma e serenidade. Agradeço por tudo que fizeste por mim.

Aos amigos Igor Catão, Lucas Vitali, Rodrigo Salinas e Victor Ferreira, pela convivência na UFSC, pelas horas de estudo e descontração. Vocês tornaram esse período um dos melhores já vividos por mim.

Aos amigos que ganhei no PET/ECV, que deram minha primeira oportunidade de aprendizado fora da sala de aula.

Ao meu orientador, Alexandre Hering Coelho, Dr., por ser tão dedicado com o que faz, um professor exemplar para a Engenharia Civil da UFSC. Pela paciência, ajuda e disponibilidade nos últimos meses.

À equipe do CEPED, onde pude aprender sobre defesa civil, área que carece de investimentos e incentivos a pesquisa.

À equipe do LabTrans/UFSC, por todo aprendizado nesta reta final da graduação. Pela convivência diária neste período de estágio.

RESUMO

A orientação da pista de pouso e decolagem e a temperatura de referência são elementos essenciais para o projeto de um aeródromo. Estes elementos auxiliam na determinação do desempenho operacional do aeródromo. Este trabalho tem como objetivo desenvolver um método e implementá-lo em *software*, de forma a automatizar o procedimento para obtenção da orientação de pistas de pouso e decolagem e da temperatura de referência de sítios aeroportuários. Uma revisão na literatura e normas técnicas foi realizada, e uma busca por dados climáticos foi fundamental para o desenvolvimento do trabalho. O *software* foi aplicado sendo utilizadas duas fontes de dados climáticos, a NOAA e o BDMEP. Em seguida foram obtidas as orientações de pista e temperaturas de referência, por meio da interpolação dos dados, para os aeródromos existentes na área de estudo. Como resultado, o *software* permitiu a verificação das orientações de pista e as temperaturas de referência calculadas com os aeródromos de referência. O *software* para a automação do procedimento de obtenção da orientação de pistas de pouso e decolagem e da temperatura de referência de sítios aeroportuários pode ser utilizado como fator de comparação para outros trabalhos e também para estudos sobre processamento de dados meteorológicos.

Palavras-chave: Aeródromos, aeroportos, orientação da pista de pouso e decolagem, temperatura de referência do aeródromo, processamento de dados.

ABSTRACT

Runway orientation and reference temperature are essential elements of an aerodrome design. These elements determine the operational performance of the aerodrome. The aim of this study is to develop a software to automate the procedure for the attainment of runway orientations and reference temperature of airport sites. To that end, a literature and technical standards review was made, also a search for climate data were fundamental for the software development. The software was applied to two climate data sources, NOAA and BDMEP. Then the runway orientations and reference temperatures were obtained by data interpolation for the existing aerodromes in the study area. As a result, the software allowed the verification the runway orientations and reference temperatures calculated for the reference aerodromes. The software for the procedure of obtaining runway orientation and the reference temperature of airport sites, can be used as a comparison factor for other works and also for studies on meteorological data processing.

Key-words: Aerodrome, airports, runway orientation, aerodrome reference temperature, data processing.

Lista de Figuras

1	Configurações de pistas de pouso e decolagem.	6
2	Fluxo de ar em torno da asa, diferença de pressão e força resultante.	7
3	Resultante das forças do vento.	8
4	Método gráfico de análise dos ventos.	10
5	Tabela de valores padrões atmosféricos.	13
6	Fluxograma do método utilizado.	16
7	Página do Banco de Dados de Superfície Integrado.	19
8	Localização das estações da NOAA no estado de Santa Catarina.	20
9	Página do Banco de Dados Meteorológicos para Ensino e Pesquisa.	23
10	Formato padrão dos dados para as estações do BDMEP.	24
11	Página para consulta das estações do BDMEP.	25
12	Localização das estações do BDMEP no Estado de Santa Catarina.	26
13	Modelo de rosa dos ventos utilizada.	30
14	Arquivo padrão dos Aeródromos.	32
15	<i>Layout</i> dos aeródromos de referência.	34
16	Superfície interpolada das orientações de pista para as estações do NOAA pelo método IDW.	40
17	Superfície interpolada das orientações de pista para as estações do NOAA pelo método RST.	40
18	Superfície interpolada das temperaturas de referência para as estações do NOAA pelo método IDW.	41
19	Superfície interpolada das temperaturas de referência para as estações do NOAA pelo método RST.	41
20	Superfície interpolada das orientações de pista para as estações do BDMEP pelo método IDW.	43
21	Superfície interpolada das orientações de pista para as estações do BDMEP pelo método RST.	43
22	Superfície interpolada das temperaturas de referência para as estações do BDMEP pelo método IDW.	44
23	Superfície interpolada das temperaturas de referência para as estações do NOAA pelo método RST.	44
24	Fator de utilização para as orientações de pista na estação 838990 - Hercílio Luz (NOAA).	45

25	Fator de utilização para as orientações de pista na estação 83897 - Florianópolis (BDMEP).	46
26	Fator de utilização para as orientações de pista na estação 838830 - Chapecó (NOAA).	46
27	Fator de utilização para as orientações de pista na estação 83883 - Chapecó (BDMEP).	47
28	Superfície interpolada das orientações de pista das estações do NOAA pelo método IDW para os aeródromos de referência.	48
29	Gráfico da relação entre as orientações observadas e as calculadas pelo <i>software</i> para os dados da NOAA pelo método IDW.	50
30	Superfície interpolada das orientações de pista das estações do NOAA pelo método RST para os aeródromos de referência.	51
31	Gráfico da relação entre as orientações observadas e as calculadas pelo <i>software</i> para os dados da NOAA pelo método RST.	51
32	Superfície interpolada das orientações de pista das estações do BDMEP pelo método IDW para os aeródromos de referência.	52
33	Gráfico da relação entre as orientações observadas e as calculadas pelo <i>software</i> para os dados da BDMEP pelo método IDW.	53
34	Superfície interpolada das orientações de pista das estações do BDMEP pelo método RST para os aeródromos de referência.	53
35	Gráfico da relação entre as orientações observadas e as calculadas pelo <i>software</i> para os dados da BDMEP pelo método RST.	54
36	Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do NOAA mais próxima, pelo método IDW.	56
37	Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do NOAA mais próxima, pelo método RST.	57
38	Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do BDMEP mais próxima, pelo método IDW.	57
39	Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do BDMEP mais próxima, pelo método RST.	58
40	Superfície interpolada das temperaturas de referência das estações da NOAA pelo método IDW para os aeródromos de referência.	60
41	Superfície interpolada das temperaturas de referência das estações da NOAA pelo método RST para os aeródromos de referência.	60
42	Superfície interpolada das temperaturas de referência das estações da BDMEP pelo método IDW para os aeródromos de referência.	61
43	Superfície interpolada das temperaturas de referência das estações da BDMEP pelo método RST para os aeródromos de referência.	61

Lista de Tabelas

1	Posição dos dados de interesse nos arquivos padrões do Banco de Dados de Superfície Integrado da NOAA.	20
2	Estações da NOAA no estado de Santa Catarina.	21
3	Completeness das estações consideradas da NOAA no estado de Santa Catarina.	22
4	Estações meteorológicas do BDMEP no Estado de Santa Catarina.	25
5	Completeness das estações do BDMEP consideradas no Estado de Santa Catarina	25
6	Aeródromos de referência.	33
7	Temperatura de referência para os aeródromos considerados como de referência.	34
8	Estrutura das pastas necessárias para o uso do <i>software</i>	35
9	Resultados para as estações do NOAA	39
10	Resultados para as estações do BDMEP	42
11	Resultado das interpolações para os aeródromos de referência.	49
12	Informações e cálculos para o Aeroporto Internacional Hercílio Luz.	55
13	Informações e cálculos para o Aeroporto Olavo Cecco Rigon.	55
14	Resultado das interpolações para temperatura de referência nos aeródromos.	59
15	Informações e cálculos para o Aeroporto Internacional Hercílio Luz.	62
16	Informações e cálculos para o Aeroporto Ministro Victor Konder.	62

Sumário

1	Introdução	1
1.1	Justificativas	2
1.2	Objetivos	2
1.2.1	Objetivo geral	2
1.2.2	Objetivos específicos	2
2	Revisão bibliográfica	4
2.1	A configuração dos aeroportos	4
2.1.1	A configuração geométrica da pista de P&D nos aeroportos	5
2.1.2	Orientação da pista de pouso e decolagem	6
2.1.3	Comprimento da pista de pouso e decolagem	11
2.2	Processamento de dados meteorológicos	14
3	Método	16
3.1	Busca e análise de dados meteorológicos (vento e temperatura)	17
3.1.1	Dados provenientes da NOAA	18
3.1.2	Dados provenientes do INMET	23
3.2	Estudo de componentes de <i>software</i> auxiliares para o desenvolvimento	26
3.3	Desenvolvimento do <i>software</i>	28
3.4	Busca e organização de dados de referência	31
3.5	Uso do <i>software</i> para geração de resultados	35
3.6	Comparação e validação	37
4	Resultados	38
4.1	Apresentação dos resultados obtidos pelo <i>software</i>	38
4.2	Comparação dos resultados com orientação de pistas existentes	48
4.3	Comparação dos resultados obtidos para temperatura de referência	58
5	Conclusões	63
	Referências	65
A	Arquivo de definição dos pontos centrais das células	67
B	Script 00: Seleciona estações do NOAA	68

C	<i>Script 01</i> : Baixa dados das estações meteorológicas do NOAA	70
D	<i>Script 02</i> : Realiza todos os cálculos	72
E	<i>Script 03</i> : Cria <i>shapfile</i> dos resultados das estações	86
F	<i>Script 04</i> : Seleciona os aeródromos da área de estudo e gera um <i>shapfile</i>	87
G	<i>Script 05</i> : Interpola os dados e determina a orientação e temperatura para os aeródromos de referência	90
H	<i>Script 06</i> : Compara os valores obtidos pelo <i>software</i> com os observados	94

1 Introdução

Para o projeto de um novo aeroporto ou de uma nova pista de pouso e decolagem, é necessário a realização de um estudo de viabilidade. Esse estudo deve incluir o cálculo da melhor orientação de pista, e também da determinação da temperatura de referência para o aeródromo.

Conforme Young e Wells (2011), as pistas de pouso e decolagem (P&D) são instalações de grande importância em um aeródromo e influenciam diretamente na capacidade de voos do aeroporto. (HORONJEFF et al., 2010)

A influência desses fatores pode ser causada pela má orientação das pistas, nas quais, em momentos que o vento de través está elevado, algumas aeronaves são impossibilitadas de realizar pousos ou decolagens.

Horonjeff et al. (2010) e Young e Wells (2011) descrevem um procedimento para determinar a melhor orientação de pista de maneira manual, utilizando papel e caneta para desenhar a rosa dos ventos e uma planilha eletrônica para os cálculos. Ainda, sugerem que sejam utilizadas séries históricas de dados de vento e temperatura dos 10 anos anteriores, para aumentar a precisão da escolha. Para isso, a *Advisory Circular* de número 150/5300-13A (FAA, 2014), cita o *National Oceanic and Atmospheric Administration* (NOAA) como fonte de dados.

Efetuar esse procedimento de forma manual é demorado e apresenta um grande risco de falhas no desenvolvimento. Para facilitar e agilizar o processo de projeto, atualmente existem *softwares* comercializados com a intenção de auxiliar os projetistas de aeroportos, que calculam a melhor orientação da pista em função dos ventos predominantes no local.

O WindRose PRO3, *software* mais completo disponível no mercado atualmente, é capaz de determinar a melhor orientação de pista, através de fontes de dados variadas, inclusive os dados da NOAA. Possui módulos para trabalhar dados de vento e exibi-los em uma rosa dos ventos, em forma de gráfico ou em uma tabela.

De forma simplificada, o WRPLOT View interpreta também os dados da NOAA e de outras fontes, e os analisa. Seus resultados são exclusivos para análise de séries históricas de vento, de forma genérica. Com eles, é possível gerar gráficos de intensidade e rosas dos ventos com o comportamento do vento para os dados analisados.

Para a temperatura de referência, o processo é feito através de uma sequência de cálculos, em que uma série histórica deve ser considerada.

Entretanto, ao fazer a escolha da localização do sitio aeroportuário, nem sempre estão disponíveis dados climáticos para uso nos cálculos, tanto para a orientação de pista de pouso e decolagem, como para a temperatura de referência do aeródromo. É necessário que esses dados sejam estimados.

Este trabalho utiliza dados meteorológicos a fim de gerar superfícies interpoladas das informações de vento e temperatura. A partir das superfícies interpoladas, são estimadas a melhor orientação de

pista e pouso e decolagem e a temperatura de referência do aeródromo.

1.1 Justificativas

Atualmente, diferente de quando ocorreu o primeiro voo de uma empresa de aviação civil, são realizadas mais de 50 milhões de viagens por ano, sendo que entre 2003 e 2008 houve um aumento de 10% nesse índice (McKinsey & Company, 2010).

Conforme consta em ANAC (2018), o Brasil registrou um cenário de crescimento econômico, e foi marcado pela aprovação da Resolução nº 400 da Agência Nacional de Aviação Civil (ANAC), de 13 de dezembro de 2016, que desregulamentou a franquia de bagagens despachadas nos voos. Além disso, foi registrada uma demanda doméstica 3,2% maior que o ano anterior.

Devido ao aumento na demanda para o transporte aéreo, atualmente o governo tem investido na privatização e concessão de estruturas aeroportuárias. O objetivo desse processo é atender às novas demandas, realizando reformas e construções de novos aeroportos.

Assim, com as pistas de pouso e decolagem sendo elementos determinantes da capacidade de um aeroporto, seu projeto é de grande importância para garantir um bom desempenho do aeródromo.

1.2 Objetivos

1.2.1 Objetivo geral

O objetivo geral deste trabalho é o desenvolvimento e a validação de um software para automatizar o procedimento de determinação da orientação das pistas de P&D, bem como a determinação da temperatura de referência do aeródromo, por meio da análise e processamento sistemático de dados climáticos.

1.2.2 Objetivos específicos

- Realizar um estudo na literatura técnica a respeito das características envolvidas com o projeto de aeródromos relacionadas com fatores climáticos.
- Investigar quais os recursos de *software* disponíveis atualmente, capazes de auxiliar no desenvolvimento do *software* proposto.
- Identificar qual é o grau de automação possível de ser alcançado em relação ao fluxo de dados, conforme a disponibilidade e as características desses dados.
- Estudar e implementar formas de visualização dos resultados que facilitem a sua análise.

- Organizar dados para serem utilizados como referência para uma comparação com os resultados obtidos com o uso do *software* desenvolvido, e sistematizar o procedimento de comparação (validação).
- Avaliar a usabilidade do *software* desenvolvido e apresentar os resultados da validação.

2 Revisão bibliográfica

A revisão bibliográfica aborda características da configuração de sítios aeroportuários, e para o cálculo da orientação de pistas de pouso e decolagem e da temperatura de referência.

2.1 A configuração dos aeroportos

Segundo Young e Wells (2011), os aeroportos podem ser tão complexos como uma cidade, e são constituídos por suas instalações físicas e sistemas. São pontos de convergência dos meios de transporte, onde os passageiros e cargas chegam através de diferentes modos e seguem para o transporte aéreo, e de divergência no caso contrário. São projetados para atender aeronaves, passageiros e cargas, e cada um, com um conjunto de componentes específicos do aeroporto que irão atendê-los.

Young e Wells (2011) colocam, os componentes de um aeroporto são divididos em duas categorias: lado ar e lado terra. Os componentes do lado ar são projetados para atender às aeronaves no aeroporto, nas operações de pouso, decolagem, estacionamento e manobras. Os componentes do lado terra, se subdividem em componentes do terminal e de acesso ao aeroporto. São projetados para atender veículos, passageiros com suas bagagens e cargas.

Ainda conforme Young e Wells (2011), o aeroporto é projetado para atender a uma demanda específica de passageiros e tipo de aeronave. As instalações devem facilitar a movimentação dos usuários, junto com suas bagagens, e também a movimentação das aeronaves.

As instalações de maior relevância em um aeroporto são:

- Pistas de pouso e decolagem (P&D);
- Pistas de táxi;
- Pátios de estacionamento de aeronaves;
- Equipamentos de auxílio à navegação;
- Sistemas de iluminação e de sinalização horizontal e vertical.

Young e Wells (2011) acrescentam que, aeroportos de pequeno porte contam, geralmente, com infraestruturas modestas com apenas uma única pista de pouso e decolagem. Além disso, não possuem pistas de táxi paralelas e contam com pequenos pátios para estacionamento de aeronaves. Já os aeroportos de grande porte, podem possuir múltiplas pistas de P&D e de táxi, além de um avançado sistema de iluminação e sinalização.

Horonjeff et al. (2010) afirmam que, devido à grande variedade de tipos de aeronaves, principalmente pela variação das dimensões, projetar um aeroporto que comporte a todas aeronaves, se

torna uma tarefa difícil. E essas variações impactam diretamente em quais aeronaves podem operar em determinado sítio. Como o peso das aeronaves, que afeta qual será a espessura do pavimento da pista de P&D, das pistas de táxi e do pátio de aeronaves.

Em consequência dessa grande variação nas características das aeronaves, autoridades nacionais e internacionais elaboraram manuais e guias, com o objetivo de auxiliar no projeto e no planejamento da operação de aeroportos. A Agência Nacional de Aviação Civil (ANAC) é a autoridade nacional responsável por normatizar e supervisionar a atividade de aviação civil. A *Federal Aviation Administration* (FAA) é responsável pelo regulamento da aviação civil dos Estados Unidos, e a *International Civil Aviation Organization* (ICAO), é uma agência especializada das Nações Unidas, que desenvolvem os princípios e técnicas de navegação aérea internacional.

Segundo Ashford, Mumayiz e Wright (2011), o *layout* de um aeroporto deve se adequar à forma e área disponível, além de satisfazer os requisitos operacionais de cada aeronave pertencente ao conjunto de aeronaves que atuará no aeroporto. É preciso conter tantas pistas quantas forem necessárias para atender à demanda de tráfego aéreo esperado. Também é importante que estejam orientadas de modo a tirar o maior proveito dos ventos predominantes do local, o que será abordado na Seção 2.1.2, e direcionadas para longe de obstáculos.

De acordo com Kazda e Caves (2015), a construção de um novo aeródromo ou a ampliação de um existente, representa um grande investimento, e é necessário que se dedique todo tempo possível para o desenvolvimento do projeto. A localização do aeroporto e a orientação do sistema de pistas deve facilitar o desenvolvimento a longo prazo e apresentar flexibilidade para uma futura expansão. Deve-se dar atenção aos dispositivos do lado ar, mas também à localização do terminal de passageiros dentro do sítio, sua expansão futura e os acessos ao aeroporto.

2.1.1 A configuração geométrica da pista de P&D nos aeroportos

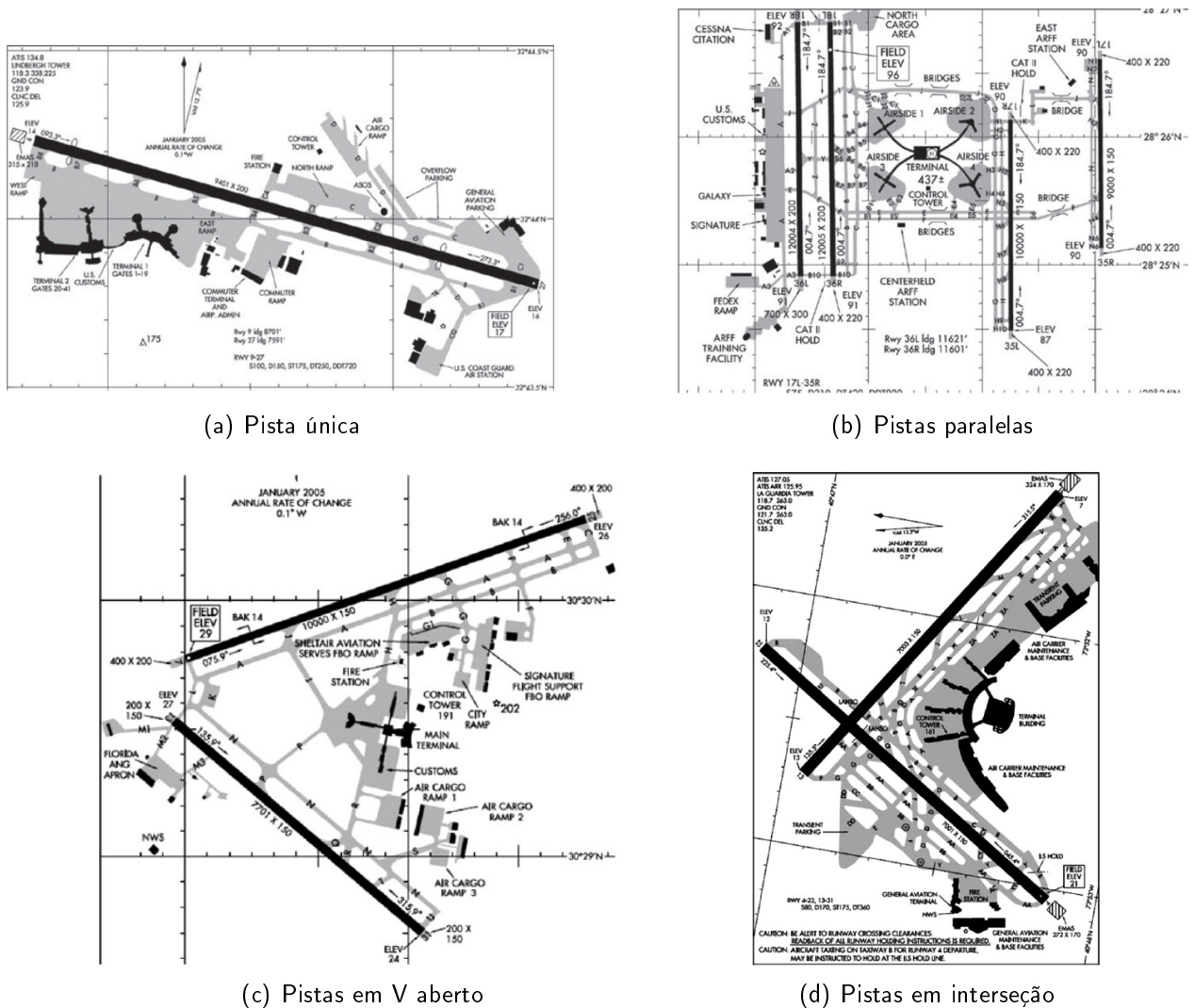
De acordo com Ashford, Mumayiz e Wright (2011), não existem regras para definir o *layout* do aeroporto. Cada local terá um projeto específico e sua configuração dependerá diretamente da quantidade e orientação das pistas de P&D, do número de pistas de táxi, tamanho e formato do pátio, área do sítio disponível, entre outros.

Segundo Horonjeff et al. (2010), a pista é uma área retangular dentro do aeroporto, preparada para receber a carga de pousos e decolagens. O aeroporto pode dispor de uma ou mais pistas de P&D, que são posicionadas, orientadas e configuradas de maneira a garantir o uso seguro e eficiente do aeroporto. Os principais pontos que influenciam a localização e orientação das pistas de pouso e decolagem são as condições climáticas do local, a direção predominante do vento, e a topografia do local e do entorno.

Desta forma, a configuração das pistas pode se dar de forma variada e refere-se à quantidade e a orientação que terão em relação umas com as outras. Essa configuração pode ser de uma única pista,

Figura 1.a, é utilizada em locais onde os ventos predominam em uma só direção. Pistas paralelas, Figura 1.b, atendem a uma maior demanda, ainda com os ventos dominantes em um sentido. As pistas em V aberto e em interseção, como mostram as Figuras 1.c e 1.d, buscam eficiência para os lugares com maiores ocorrência de ventos de través.

Figura 1: Configurações de pistas de pouso e decolagem.



Fonte: Horonjeff et al. (2010)

2.1.2 Orientação da pista de pouso e decolagem

A orientação da pista de pouso e decolagem é dada pela direção predominante dos ventos no sítio em relação ao norte magnético. Segundo Abdalla, Osman e Mohamed (2011), o comportamento do vento no local é o ponto mais crítico e preocupante para os projetistas durante a determinação da orientação da pista de P&D e escolha do sítio aeroportuário.

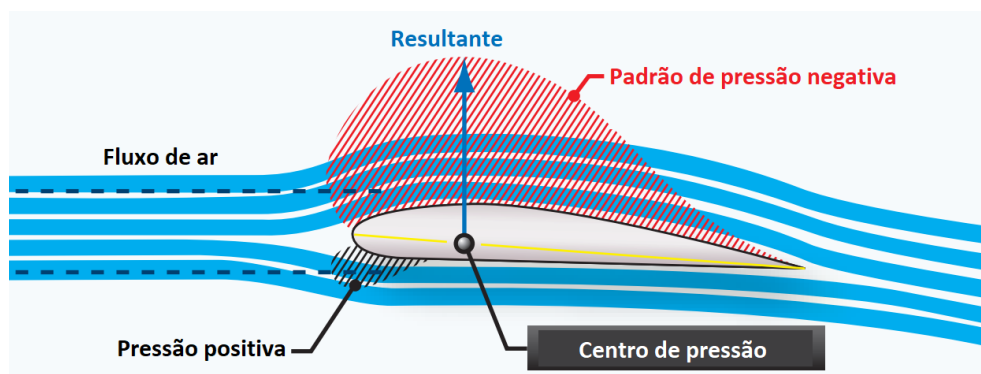
Também, segundo Abdalla, Osman e Mohamed (2011), a direção do vento predominante não só determina a orientação das pistas de pouso e decolagem, como também a localização dos outros componentes do aeródromo. Como o primeiro passo no projeto de um aeroporto, relaciona-se com a localização das pistas de P&D, as demais instalações serão distribuídas no entorno das pistas de P&D.

Conforme Young e Wells (2011) afirmam, é possível que a pista seja a instalação mais importante do aeródromo, pois sem uma pista planejada de forma adequada, a aeronave que irá operar neste aeroporto seria incapaz de pousar ou decolar com segurança. Quando os irmãos Wright fizeram seu primeiro voo em 1903, não existiam pistas que facilitassem a operação. Mas certas condições levaram diretamente à orientação das pistas que temos hoje.

Ainda segundo Young e Wells (2011), os irmãos Wright sabiam que, as aeronaves com asa fixa, precisariam do máximo fluxo de ar sobre as asas para atingir o voo. Perceberam que a melhor direção para decolar a aeronave era contra o vento, pois esta permitia que a aeronave conseguisse a quantidade necessária de fluxo de ar sobre as asas com uma menor velocidade no solo e menor distância de decolagem. Para o pouso, da mesma forma que na decolagem, a direção mais segura para a operação era contra o vento, pois mesmo que com uma menor velocidade, o fluxo maior de ar sobre as asas aumenta a sustentação da aeronave. Como resultado, as pistas são tipicamente orientadas na direção dos ventos predominantes no local.

Conforme o Manual Técnico de Manutenção de Aviação - Estrutura, volume I da FAA (2018), as propriedades do ar afetam o controle e a performance da aeronave. O ar é um fluido gasoso. Pode ser comprimido ou expandido, alterando, assim, a sua pressão. A pressão atua sobre o que está em contato com o ar, neste caso, as asas da aeronave. Com o movimento da aeronave, a ação que a asa exerce sobre o ar, gera um aumento na pressão sob a asa e uma redução da pressão acima da asa, conforme ilustrado na Figura 2.

Figura 2: Fluxo de ar em torno da asa, diferença de pressão e força resultante.



Fonte: FAA (2018)

Ainda segundo o Manual Técnico de Manutenção de Aviação - Estrutura volume I da FAA (2018), a diferença de pressões resulta em uma força resultante na asa, chamada de força de sustentação, fazendo com que ela se eleve. Na Figura 2 também é possível ver esta resultante.

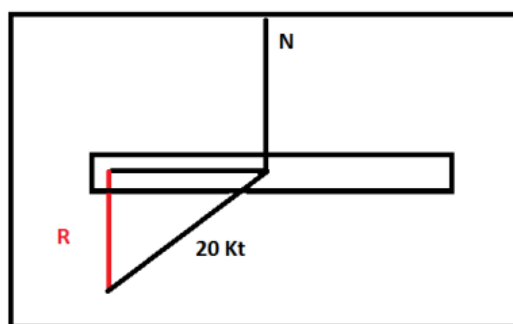
Conforme o Manual Técnico de Manutenção de Aviação - Estrutura volume I da FAA (2018), o ar é feito de partículas de matéria, e esta sofre com a variação da temperatura. Com a variação da temperatura, a densidade do ar se altera e afeta a ação que a asa exerce sobre o ar. Caso a temperatura se eleve, a densidade terá um comportamento inversamente proporcional e diminuirá. Esta diminuição da densidade influencia na força de sustentação, pelo fato da pressão exercida sob a asa diminuir.

A redução da sustentação da aeronave devido à diminuição da pressão deve ser compensada com o aumento da velocidade da aeronave, precisando assim, de uma distância maior em solo para atingir a sustentação necessária.

Também, Young e Wells (2011) afirmam que, assim como é mais recomendado que a aeronave decole ou pouse em direção contrária ao vento, é menos recomendado, e às vezes altamente inseguro, decolar ou pousar com vento perpendicular à trajetória do avião, isto é, com vento de través. Aeronaves menores e mais leves tendem a ser mais sensíveis aos ventos de través do que as maiores.

Para entender a interferência do vento cruzado nas aeronaves, Pamplona e Alves (2013) exemplificaram em um desenho (Figura 3). Supondo-se um vento a 20 nós de velocidade, que incida na pista em um ângulo de 10° , teremos uma resultante, R, que será dada por $R = 20 \cdot \sin 10^\circ$. Para o exposto, o valor calculado para a resultante é de 3,4 nós. Esta resultante é considerada como o vento de través.

Figura 3: Resultante das forças do vento.



Fonte: Pamplona e Alves (2013)

Segundo a *Advisory Circular* de número 150/5300-13A - FAA (2014), as pistas devem ser posicionadas e orientadas para prover maior segurança e eficiência. Também devem estar orientadas na direção dos ventos predominantes, sendo o vento um fator que influencia não só na orientação, mas também no número de pistas que o aeroporto vai possuir. Isto deve ao fato de que, quando

a distribuição dos ventos é dada por mais de uma direção predominante e o objetivo é também atender aeronaves menores, ou seja, mais sensíveis ao vento de través, é necessário uma segunda pista para atender essas aeronaves.

No Brasil, a ANAC disponibiliza o Regulamento Brasileiro da Aviação Civil (RBAC) nº 154 (ANAC, 2019), que orienta sobre o número de pistas de pouso e decolagem do aeródromo. É determinado pelo fator de utilização do aeródromo, que deve ser no mínimo 95% nos casos de pista simples. Quando menor, é recomendada a existência de uma segunda pista. O fator de utilização significa, em porcentagem, o tempo que a pista de P&D consegue operar sem que tenha sua capacidade limitada pelo vento de través.

Ainda, o Anexo 14 da ICAO (2018) e o RBAC nº 154 (ANAC, 2019) estabelecem que, os limites de vento de través para pouso e decolagem são:

- 37 km/h (20 nós), no caso de aeronaves cujo comprimento básico de pista, que é explicado na Seção 2.1.3 deste trabalho, é maior ou igual a 1.500 m, exceto quando houver, com certa frequência, uma baixa ação de frenagem na pista devido a um coeficiente de atrito longitudinal insuficiente, quando, então, deve ser assumido um componente de vento de través que não exceda 24 km/h (13 nós);
- 24 km/h (13 nós), no caso de aeronaves cujo comprimento básico de pista é maior ou igual a 1.200 m e menor que 1.500 m;
- 19 km/h (10 nós), no caso de aeronaves cujo comprimento básico de pista for menor que 1.200 m.

Para determinar a orientação da pista de P&D, segundo Young e Wells (2011), é necessário que se identifique dentre as aeronaves que irão operar no aeroporto, aquela que é mais sensível ao vento de través. E esta informação pode ser encontrada no manual de operação de cada aeronave, comparando o comprimento básico de pista com os limite apresentados pelo Anexo 14 da ICAO (2018) e o RBAC nº 154 (ANAC, 2019). Ainda, a *Federal Aviation Administration* orienta que para ser considerada crítica, a aeronave deve realizar pelo menos 500 operações por ano. Assim, definida a aeronave crítica, o vento de través máximo permitido é identificado.

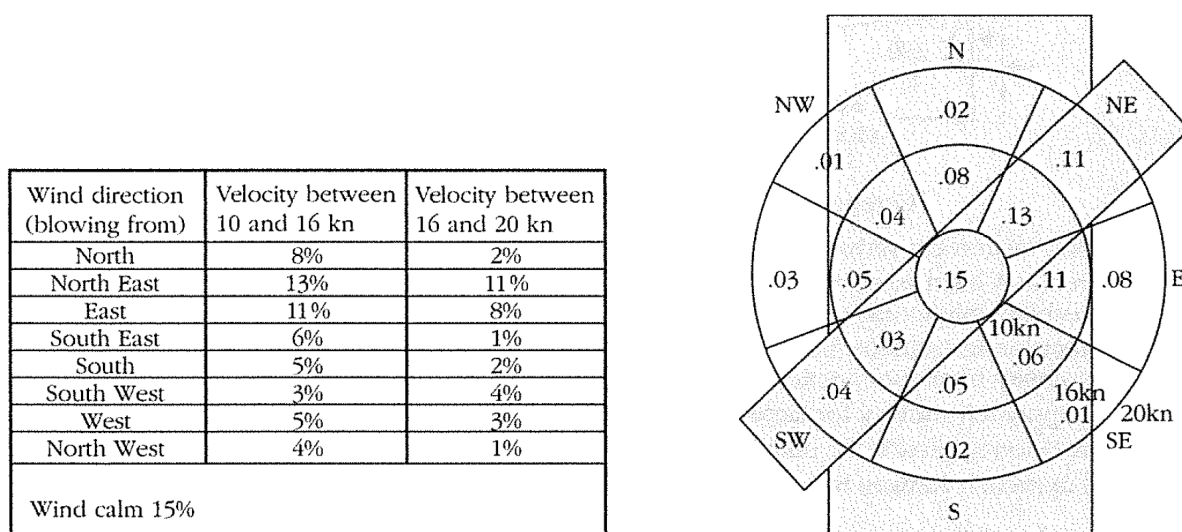
Segundo o Apêndice G do RBAC nº 154 (ANAC, 2019), os dados precisam ser os mais confiáveis, de maneira que sejam representativos para a área estudada. Também, a *Advisory Circular* 150/5325-4B (FAA, 2005) sugere a utilização de uma série histórica para que o estudo tenha um resultado seguro.

Segundo Young e Wells (2011), para a análise dos ventos, é medida sua velocidade e direção nos aeroportos e pontos de interesse. Esses dados, são historicamente registrados em uma base de

dados do órgão responsável pelas estações de medição. São compilados, categorizados e ilustrados com o auxílio gráfico de uma rosa dos ventos.

Conforme ilustrado na Figura 4, de Young e Wells (2011), a rosa dos ventos (Figura 4.b) representa graficamente as ocorrências de velocidades do vento e suas direções. Os anéis concêntricos representam a velocidade do vento, e os raios são sua direção. No caso exposto por Young e Wells (2011), ao centro se concentram os ventos com intensidade menor que 10 nós, sendo considerados como ventos calmos. A próxima marca, no anel intermediário, é de ventos com velocidades de 10 a 16 nós. Para o último anel, as velocidades são de 16 a 20 nós.

Figura 4: Método gráfico de análise dos ventos.



(a) Percentual de incidência dos ventos em cada faixa de velocidade e direção

(b) Rosa dos ventos

Fonte: Young e Wells (2011)

No exemplo de Young e Wells (2011) (Figura 4), os dados de vento foram ordenados na tabela da Figura 4.a. Todas as amostras dos dados estão classificados em percentual de participação, e estão de acordo com a intensidade e a orientação do vento.

Na Figura 4.b é possível identificar dois retângulos desenhados sobre a rosa dos ventos, estes que representam dois limites de vento cruzado. Sua largura é definida pelo vento de través admissível por cada uma das aeronaves. A pista Norte-Sul, para aeronaves com vento de través admissível de 16 nós. Já a pista Nordeste-Sudoeste, com vento de través admissível de 10 nós.

Essas orientações foram obtidas por meio do procedimento gráfico. Ashford, Mumayiz e Wright (2011) descrevem o procedimento que deve ser feito após ter representado os ventos na rosa dos ventos:

1. Colocar o modelo de plástico da pista, com a largura de duas vezes o vento de través permitido,

com a linha do meio passando pela origem da rosa dos ventos.

2. Usando a origem da rosa dos ventos como centro de giro, rotacionar o modelo de modo que a soma das porcentagens entre as linhas seja a máxima possível.
3. Ler a orientação obtida com o auxílio da rosa dos ventos.
4. A soma das porcentagens entre as linhas indica o fator de utilização, ou seja, o tempo que uma pista estará disponível para tal vento de través. E satisfará o requisito de vento de través.

Ashford, Mumayiz e Wright (2011) destacam a importância de se conhecerem os dados de vento obtidos, observar quais são as unidades que são apresentados. É comum que os ventos estejam orientados em relação ao norte verdadeiro. Para isso, é necessário que a orientação obtida a partir do modelo gráfico seja corrigida pela declinação magnética da localização do aeroporto, e assim obter a orientação em relação ao norte magnético.

De acordo com Young e Wells (2011), as pistas são nomeadas por sua orientação em relação ao norte magnético. A identificação da pista é chamada de designação, e é feita utilizando-se o azimute, ângulo da pista até o norte magnético, dividido por dez, arredondado para o inteiro mais próximo. Também, quando o aeródromo possui pistas paralelas, a identificação de cada pista é dada pela sua orientação mais uma letra, "L" ou "R", para pista da esquerda e pista da direita, respectivamente. No caso de uma terceira pista paralela, a central tem o acréscimo da letra "C".

2.1.3 Comprimento da pista de pouso e decolagem

O comprimento da pista de pouso e decolagem é um fator de grande importância no projeto de um aeroporto, e para isso, é necessário calcular a temperatura de referência do aeródromo. Segundo Ashford, Mumayiz e Wright (2011), a pista deve ter comprimento suficiente para atender com segurança as operações de pouso e decolagem. O comprimento é influenciado pelos seguintes fatores:

- Características das aeronaves que irão operar no aeroporto;
- Peso das aeronaves durante a decolagem e o pouso;
- Elevação do aeroporto;
- Temperatura de referência do aeródromo; e
- Declividade da pista.

Para o seu dimensionamento, a FAA disponibiliza na *Advisory Circular 150/5325-4B* (FAA, 2005), um guia para a determinação do comprimento da pista de pouso e decolagem. Para os projetos no Brasil, o RBAC nº 154 (ANAC, 2019) deve ser observado. Entretanto, ela não apresenta um procedimento para obtenção do comprimento de pista, mas indica os fatores relevantes para a determinação do projeto. São as características da aeronave escolhida como crítica, as condições climáticas (principalmente o vento e a temperatura), as características da pista (declividade e condição superficial) e a sua localização.

Como visto na Seção 2.1.2, sítios localizados em altitudes elevadas e/ou expostos a temperaturas elevadas, terão seu ar menos denso, o que implica em uma velocidade e um comprimento de pista maiores, para atingir uma mesma força de sustentação e alçar voo. Em operações de pouso, não é diferente. É necessário um comprimento de pista maior, pois a aeronave fará a aproximação em uma velocidade mais elevada para suprir a mesma força de sustentação.

Segundo RBAC nº 154 (ANAC, 2019), o comprimento básico de pista da aeronave significa o comprimento mínimo de pista necessário para que a aeronave possa decolar com seu peso máximo de decolagem certificado, no nível do mar, em condições atmosféricas normais, com o ar parado e em uma pista sem declividade.

O Anexo 14 da ICAO (2018) recomenda que, o comprimento não deve ser menor que o maior comprimento calculado utilizando-se as correções indicadas na mesma norma. Esses fatores de correção são para considerar a diferença de altitude da pista em relação ao nível médio do mar, a temperatura de referência do aeroporto e a declividade longitudinal da pista.

O procedimento para a correção do comprimento deve ser executado na sequência apresentada no *Aerodrome Design Manual - Part 1: Runways* (ICAO, 2006). Para a obtenção do comprimento final, é necessário que se calcule primeiro o comprimento corrigido em função da altitude. Onde, para cada 300 metros de elevação, em relação ao nível do mar, é acrescentado 7 por cento. Como mostrado na Equação 1.

$$C_a = \left(C_b \times \frac{7}{100} \times \frac{H}{300} \right) + C_b \quad (1)$$

Onde:

- C_a : Comprimento corrigido pela altitude (m).
- C_b : Comprimento básico para a aeronave (m).
- H : Altitude (m).

Em seguida, deve ser feita a correção pela temperatura. Esse cálculo é feito em função da temperatura de referência do aeródromo e a temperatura padrão atmosférica para a sua altitude. A Figura 5 apresenta uma tabela com as temperaturas padrões referentes a cada altitude. Então, é

acrescentado 1 por cento a cada grau da diferença de temperatura entre a de referência e a padrão. O cálculo é feito através da Equação 2.

Figura 5: Tabela de valores padrões atmosféricos.

Altitude (m)	Temperature (Centigrade)	Pressure (Kg/m ³)
0	15.00	1.23
500	11.75	1.17
1 000	8.50	1.11
1 500	5.25	1.06
2 000	2.00	1.01
2 500	-1.25	0.96
3 000	-4.50	0.91
3 500	-7.75	0.86
4 000	-10.98	0.82
4 500	-14.23	0.78
5 000	-17.47	0.74
5 500	-20.72	0.70
6 000	-23.96	0.66

Fonte: ICAO (2006)

$$C_t = \left(C_a \times \frac{1}{100} \times (T_r - T_p) \right) + C_a \quad (2)$$

Onde:

- C_t : Comprimento corrigido pela temperatura (m).
- C_a : Comprimento corrigido pela altitude (m).
- T_r : Temperatura de referência do aeródromo (°C).
- T_p : Temperatura de padrão atmosférica dada para a altitude do aeródromo (°C).

Caso essas correções ultrapassem 35%, deverá ser feito um estudo específico para o aeródromo. Se isso não ocorrer, deverá ainda ser aplicada uma correção para a declividade. Caso o comprimento básico for maior ou igual a 900 metros, o comprimento corrigido pela temperatura deverá agora ser aumentado em 10% para cada 1% de declividade da pista de pouso e decolagem.

É imprescindível que se calcule a temperatura de referência. Para isso, a *Advisory Circular* 150/5325-4B (FAA, 2005), o Anexo 14 da ICAO (2018) e o RBAC nº 154 (ANAC, 2019), apresentam um mesmo procedimento para se obter a temperatura de referência do aeroporto. Deve ser considerada uma amostra com alguns anos, onde a temperatura de referência é a média das temperaturas

máximas diárias durante o mês mais quente do ano. O mês mais quente é determinado com a temperatura mensal média. A temperatura para o cálculo é medida por meio de um termômetro de bulbo seco.

2.2 Processamento de dados meteorológicos

Como visto, é indispensável o uso de dados climáticos para o projeto de um aeroporto, tanto para definir a orientação da pista de P&D, como também para a determinação da temperatura de referência do aeródromo. Esses dados, fornecidos por estações meteorológicas, podem ser obtidos de maneira bruta, ou seja, em meio a diversas outras informações. Ou então, de uma maneira mais objetiva, que a variável desejada é informada separadamente.

De acordo com Vianna et al. (2017), os bancos de dados meteorológicos são projetados para gerenciar diversas variáveis meteorológicas, como por exemplo precipitação pluviométrica, temperatura do ar, entre outros. Esses dados são gerados por cada uma das estações meteorológicas, localmente, ou ainda, por sistemas de captação remota.

Vianna et al. (2017) colocam que, com o desenvolvimento da tecnologia, os sistemas de posicionamento global (GPS, do inglês *Global Positioning System*) e os sistemas de informações geográficas (SIG) se tornam mais avançados, e conseqüentemente, o seu uso mais comum para analisar dados meteorológicos através de superfícies contínuas. As superfícies são elaboradas através do processamento dos dados das estações, ou por meio de regressões ou interpolações dos dados.

Segundo Vianna et al. (2017), o processamento e o estudo de superfícies contínuas de variáveis meteorológicas possibilitam a identificação de áreas expostas a riscos climáticos, e também, fazer a previsão de tempo e estudos climáticos. É citado ainda, que o NOAA, o INMET, e outros órgãos, utilizam estes mapas para representar variações de temperatura e precipitação. Colocam ainda, que para mapas gerados por interpolação ou por equações de regressão, é fundamental que se conheça o processo de geração dos dados.

Bagio et al. (2015) afirmam que, os métodos de interpolação possibilitam a projeção estatística de um conjunto de dados georreferenciados, para a obtenção de superfícies contínuas. Ainda, os autores levantaram os métodos de interpolação mais utilizados para variáveis climáticas como temperatura e precipitação são: *Kriging*, *Inverse Distance Weighting* (IDW, Inverso do Quadrado por um Peso) e *Spline*.

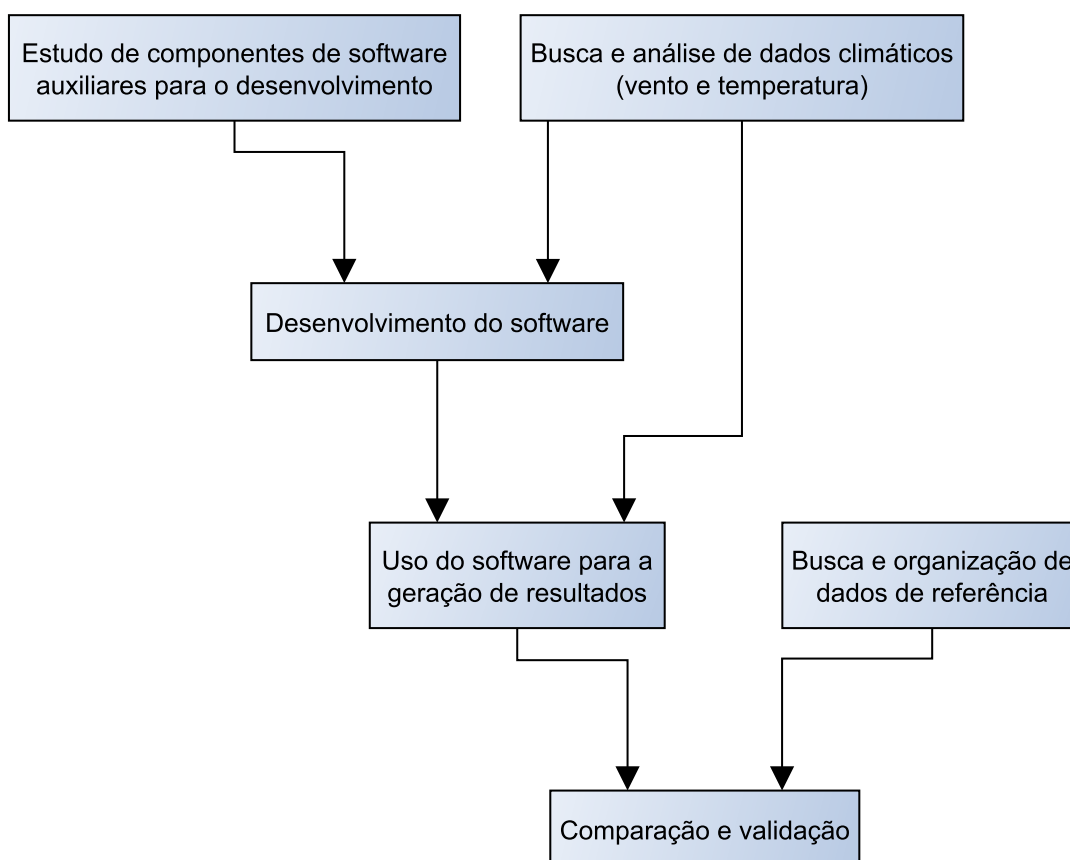
Conforme Barbosa (2006), o *Kriging* é um método de interpolação que permite estimar um valor desconhecido, a partir de um conjunto de dados disponíveis. Em Mello et al. (2003), o Inverso do Quadrado da Distancia é exemplificado como, quanto mais longe o ponto a ser verificado estiver do ponto amostral, maior será a diferença entre eles. E ainda, a potência quadrática pode variar, indicando o quão importante é a variação da distância para a variável estudada. Para o método

Spline, segundo Mitas e Mitasova (1999), é formado por curvas interpoladas e estas devem passar pelos pontos conhecidos e suavizar as curvas geradas entre eles.

3 Método

Este capítulo tem como objetivo descrever o método utilizado neste trabalho. O fluxograma, apresentado na Figura 6, mostra as etapas desenvolvidas, que são explicadas em detalhes na sequência deste capítulo.

Figura 6: Fluxograma do método utilizado.



3.1 Busca e análise de dados meteorológicos (vento e temperatura)

Para o desenvolvimento do presente trabalho, foi necessário encontrar fontes de dados meteorológicos. Devido à grande importância da escolha da orientação da pista de pouso e decolagem e da temperatura de referência, a fonte de dados precisa ser um órgão devidamente reconhecido pelo fornecimento de dados meteorológicos.

O Anexo 14 da ICAO (2018) apresenta uma recomendação de como deve ser a seleção dos dados a serem utilizados. Segundo a publicação, os dados devem se basear em estatísticas confiáveis de distribuição para o maior período possível, e preferencialmente, não menos de 5 anos, com no mínimo 8 observações em um mesmo dia, espaçadas por um mesmo intervalo de tempo.

A busca por dados deve ser direcionada aos objetivos do trabalho. Para o procedimento de obtenção da melhor orientação da pista de pouso e decolagem, os dados necessários para o cálculo são relativos ao vento. Para isso, são utilizados a velocidade e direção do vento (conforme apresentado na Seção 2.1.2). No cálculo da temperatura de referência, é necessária a temperatura de bulbo seco medida em diferentes horários.

Para o presente trabalho, a busca foi restringida aos dados de vento e de temperatura, pois ao procurar dados provenientes de estações de monitoramento climático, é possível encontrar registros de pluviometria, de umidade do ar, de pressão atmosférica, entre outros. Também, a área de estudo foi definida como sendo o Estado de Santa Catarina.

Assim, foi preciso encontrar fontes que disponibilizassem seus dados, e que eles pudessem ser obtidos de forma sistemática.

Para que os dados possam ser interpretados, é necessário que a fonte disponibilize também os metadados, que contêm as informações dos dados que foram obtidos. É de extrema importância, para assim ter plenos conhecimentos sobre eles e como os dados são disponibilizados, por meio de um serviço de *web service*, arquivos ".csv", ou até mesmo planilhas eletrônicas.

Ainda, é preciso identificar quais estações meteorológicas serão utilizadas para a área de estudo desejada, definir o período dos dados e verificar sua completude.

Por meio de pesquisas na *internet*, algumas opções surgiram e dentre elas, três se destacaram: a NOAA, o Instituto Nacional de Meteorologia (INMET) e a Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina (Epagri). A seguir, são apresentadas cada uma das alternativas.

- NOAA: A Administração Oceânica e Atmosférica dos Estados Unidos oferece um portal, o Centro Nacional de Dados Climáticos (NCDC), onde é possível encontrar diversos dados relacionados a meteorologia, tais como: séries históricas de tempo e clima e, também, previsões de eventos meteorológicos. Além disso, o órgão disponibiliza online os dados de milhares de estações meteorológicas, com dados de séries históricas de mais de 40 anos (NOAA, 2019).

- INMET: O instituto pertence ao Ministério da Agricultura, Pecuária e Abastecimento do governo do Brasil. Fornece informações meteorológicas à população, estas que são obtidas através de monitoramento, análise e previsão de tempo e clima. O órgão nacional possui um portal de nome Banco de Dados Meteorológicos para Ensino e Pesquisa (BDMEP), e tem o objetivo de auxiliar em atividades acadêmicas. O BDMEP disponibiliza séries históricas de estações meteorológicas no Brasil desde 1961. Temperaturas, umidade do ar, direção e velocidade do vento são exemplos das informações disponibilizadas. (BDMEP, 2019)
- Epagri: É uma empresa pública, vinculada ao Governo do Estado de Santa Catarina por meio da Secretaria de Estado da Agricultura e da Pesca (EPAGRI, 2019). Possui o Centro de Informações de Recursos Ambientais e de Hidrometeorologia de Santa Catarina (Ciram), que monitora o clima no estado por meio de diversas estações meteorológicas. Disponibiliza seus dados para atividades de ensino e pesquisa, mediante solicitação formal da instituição de ensino, e comercializa os mesmo dados para projetos desenvolvidos por empresas privadas. (EPAGRI - CIRAM, 2019)

Duas das alternativas estudadas disponibilizam os dados de forma *online* e possibilitam sua obtenção de forma sistemática, que são: a NOAA e o INMET. Estas são adotadas neste estudo.

3.1.1 Dados provenientes da NOAA

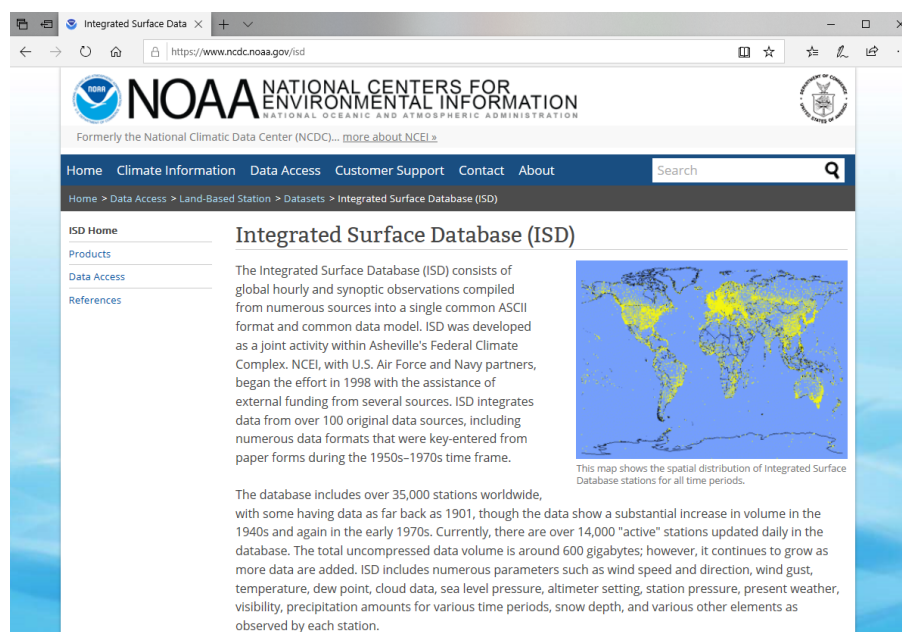
A NOAA disponibiliza o módulo *Integrated Surface Database* (ISD, Banco de Dados de Superfície Integrado) (cujo portal de acesso se encontra na Figura 7), que possui os dados das estações meteorológicas. É possível obter os dados de diferentes modos, através de uma busca por uma estação, ou até mesmo para uso em sistemas, por meio de *web services*.

Para o uso com programação, é oferecido um servidor de *File Transfer Protocol* (FTP) e um conjunto de *Application Programming Interface* (API). Para o trabalho, foi utilizado o servidor de FTP, por sua aplicação ser mais simples.

No processo de análise dos dados, a NOAA oferece uma série de documentos com informações para possibilitar a sua compreensão, os metadados. O primeiro deles, o "*isd-history.txt*", em formato de texto, dispõe de dados referentes a todas estações de forma geral. Cada estação tem atrelada à ela: código único de identificação, código do NCDC, país, estado (para estações nos Estados Unidos), código ICAO (quando a estação estiver localizada em um aeroporto), latitude, longitude, elevação e as datas de início e fim das medições.

O segundo documento de apoio, "*isd-format-document.pdf*", mostra detalhadamente o que cada informação dos arquivos das estações representa e a sua posição no arquivo. Este arquivo é muito importante, pois sem ele não seria possível a interpretação dos dados das estações.

Figura 7: Página do Banco de Dados de Superfície Integrado.



Fonte: NOAA (2019)

Por fim, os arquivos referentes a cada estação possuem um nome padrão, para que seja possível sua identificação mesmo sem abrir o arquivo. Com esses arquivos são obtidos os dados de vento e de temperatura para cada estação. Seu nome é composto da seguinte maneira:

XXXXXX-EEEE-YYYY.gz

Onde:

- XXXXXX: Código único de identificação da estação.
- EEEE: código dado pelo NCDC.
- YYYY: Ano de referência do arquivo.

Com os arquivos das estações, e as especificações contidas no documento "*isd-format-document.pdf*", foi possível identificar a posição, a unidade e a maneira como os dados de vento e de temperatura são apresentados. A Tabela 1 mostra a posição dos dados que são utilizados.

Foi preciso fazer um filtro no arquivo de dados de informações das estações de modo que somente as estações de Santa Catarina fossem consideradas. Para isso, foi utilizado o *software* QGIS, que é um Sistema de Informação Geográfica (SIG) (QGIS, 2019) livre. Também foi utilizado um arquivo do tipo *shapefile*, que possui informações da georreferenciadas acerca da geometria do estado de Santa Catarina, obtido no site¹ do IBGE, para selecionar as estações no Estado de Santa Catarina.

¹<<https://www.ibge.gov.br/geociencias/downloads-geociencias.html>>

Tabela 1: Posição dos dados de interesse nos arquivos padrões do Banco de Dados de Superfície Integrado da NOAA.

Posição	Tipo	Descrição
16 - 27	Data	Data e hora do registro*
61 - 63	Inteiro	Ângulo do vento observado (graus)
65 - 65	Texto	Tipo do dado de vento observado ("N" para medições normais)
66 - 69	Inteiro	Velocidade do vento (m/s), com fator de escala de 10
88 - 92	Inteiro	Temperatura do ar (graus Celsius), com fator de escala de 10

*Obs: A data está no formato AAAAMMDDHHmm, sendo AAAA = Ano, MM = Mês, DD = Dia, HH = Hora e mm = minutos

Após a execução filtro, foram identificadas 34 estações meteorológicas em todo o estado, listadas na Tabela 2.

Entretanto, nem todas as estações possuem dados que possibilitem sua utilização. Foram identificadas 10 estações em que os dados não estavam sendo disponibilizados. Portanto, foram consideradas 24 estações para este estudo, as quais são apresentadas na Tabela 3, juntamente com a completude dos dados de cada uma. Na Figura 8 é mostrada a localização das estações no território estadual.

Figura 8: Localização das estações da NOAA no estado de Santa Catarina.

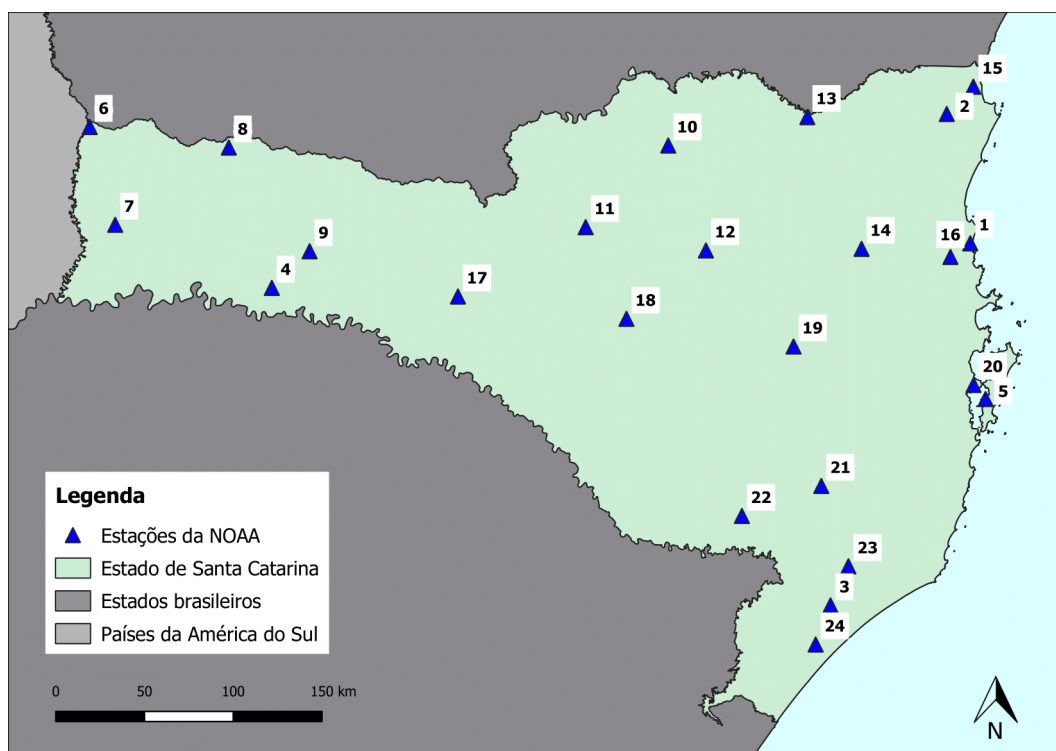


Tabela 2: Estações da NOAA no estado de Santa Catarina.

Código da estação	Nome da estação
825991	Ministro Victor Konder INTL
827981	Lauro Carneiro de Loyola
836727	Criciúma/Forquilha
838690	Irineópolis
838720	Indaial
838740	São Francisco do Sul
838830	Chapecó
838870	Campos Novos
838890	Imaribo
838910	Lages
838970	Florianópolis
838990	Hercílio Luz
839200	S. Joaquim
839230	Urussanga
839240	Laguna
869360	Dionísio Cerqueira
869370	São Miguel do Oeste
869380	Novo Horizonte
869400	Xanxerê
869420	Major Vieira
869430	Caçador
869440	Rio do Campo
869450	Rio Negrinho
869460	Indaial
869470	Itapoá
869480	Itajaí
869550	Joaçaba
869560	Curitibanos
869570	Ituporanga
869580	Florianópolis-São José
869680	Urubici
869690	São Joaquim
869700	Urussanga
869710	Araranguá

Tabela 3: Completude das estações consideradas da NOAA no estado de Santa Catarina.

	Código da estação	Nome da estação	Anos na amostra	Total de amostras
1	825991	Ministro Victor Konder INTL	5	34124
2	827981	Lauro Carneiro de Loyola	5	23458
3	836727	Criciúma/Forquilha	3	10184
4	838830	Chapecó	5	27124
5	838990	Hercílio Luz	5	39315
6	869360	Dionísio Cerqueira	4	26651
7	869370	São Miguel do Oeste	4	13580
8	869380	Novo Horizonte	4	18327
9	869400	Xanxerê	4	23313
10	869420	Major Vieira	4	27012
11	869430	Caçador	4	19719
12	869440	Rio do Campo	4	23343
13	869450	Rio Negrinho	4	25283
14	869460	Indaial	4	25166
15	869470	Itapoá	4	24902
16	869480	Itajaí	4	26483
17	869550	Joaçaba	4	23858
18	869560	Curitibanos	4	25965
19	869570	Ituporanga	4	24333
20	869580	Florianópolis-São José	4	15641
21	869680	Urubici	4	20114
22	869690	São Joaquim	4	26296
23	869700	Urussanga	4	16622
24	869710	Araranguá	4	20841

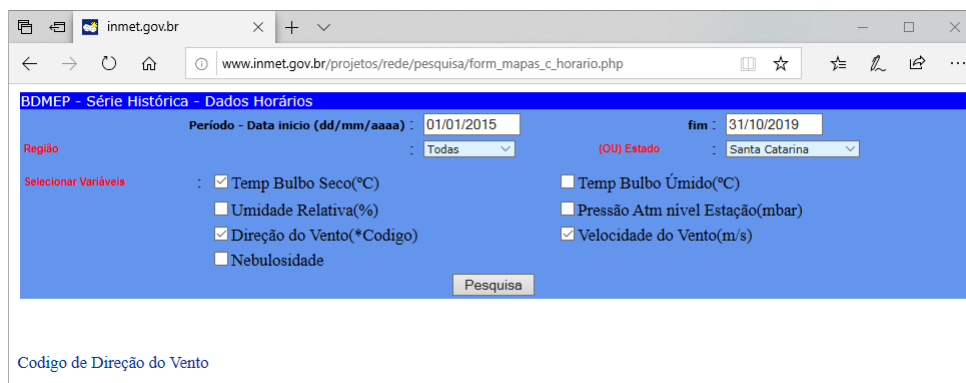
3.1.2 Dados provenientes do INMET

O BDMEP possui um portal onde são disponibilizados os dados climáticos (cujo portal de acesso pode ser visto na Figura 9.a). Neste sistema, o usuário precisa, através de um cadastro e posterior autenticação, fazer uma consulta informando quais informações esta solicitando (Figuras 9.b).

Figura 9: Página do Banco de Dados Meteorológicos para Ensino e Pesquisa.



(a) Página de acesso ao BDMEP



(b) Página para consulta dos dados

Fonte: BDMEP (2019)

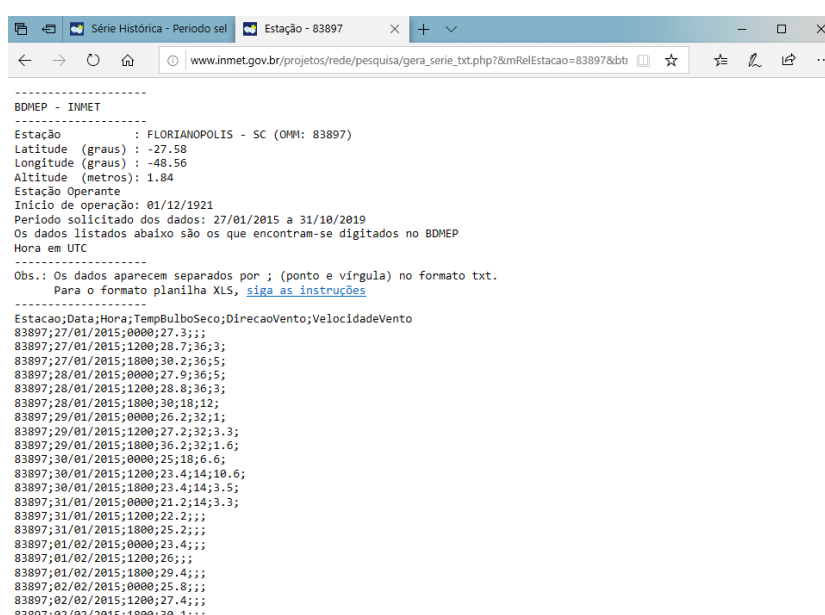
Para o uso com programação, a forma de obtenção é diferente das estações da NOAA. É necessário que o *software* interprete a página acessada, e obtenha os dados diretamente do que é mostrado no site (Esse processo está descrito na Seção 3.3).

Os dados contidos em cada pesquisa são personalizados de acordo com o solicitado. Para o

trabalho, a requisição foi para os dados relacionados à direção do vento e à temperatura de bulbo seco, que é a temperatura da atmosfera sem considerar o efeito da umidade.

No processo de análise dos dados, o BDMEP não possui uma documentação de apoio acerca do formato e informações disponíveis no portal. A Figura 10 mostra como são disponibilizados os dados de cada estação, e estes se assemelham com o formato “.csv”, em que as informações contidas nele são separadas por vírgula. Contudo, neste caso os separadores são pontos e vírgulas. Também, no cabeçalho são apresentadas informações geográficas da estação, como também, seu nome, data de início de operação e período das amostras de dados disponibilizadas.

Figura 10: Formato padrão dos dados para as estações do BDMEP.



```
-----  
BDMEP - INMET  
-----  
Estação : FLORIANOPOLIS - SC (OMM: 83897)  
Latitude (graus) : -27.58  
Longitude (graus) : -48.56  
Altitude (metros): 1.84  
Estação Operante  
Início de operação: 01/12/1921  
Período solicitado dos dados: 27/01/2015 a 31/10/2019  
Os dados listados abaixo são os que encontram-se digitados no BDMEP  
Hora em UTC  
-----  
Obs.: Os dados aparecem separados por ; (ponto e vírgula) no formato txt.  
Para o formato planilha XLS, siga as instruções  
-----  
Estacao;Data;Hora;TempBulboSeco;DirecaoVento;VelocidadeVento  
83897;27/01/2015;0000;27.3;;;  
83897;27/01/2015;1200;28.7;36;3;  
83897;27/01/2015;1800;30.2;36;5;  
83897;28/01/2015;0000;27.9;36;5;  
83897;28/01/2015;1200;28.8;36;3;  
83897;28/01/2015;1800;30;18;12;  
83897;29/01/2015;0000;26.2;32;1;  
83897;29/01/2015;1200;27.2;32;3.3;  
83897;29/01/2015;1800;36.2;32;1.6;  
83897;30/01/2015;0000;25;18;6.6;  
83897;30/01/2015;1200;23.4;14;10.6;  
83897;30/01/2015;1800;23.4;14;3.5;  
83897;31/01/2015;0000;21.2;14;3.3;  
83897;31/01/2015;1200;22.2;;;  
83897;31/01/2015;1800;25.2;;;  
83897;01/02/2015;0000;23.4;;;  
83897;01/02/2015;1200;26;;;  
83897;01/02/2015;1800;29.4;;;  
83897;02/02/2015;0000;25.8;;;  
83897;02/02/2015;1200;27.4;;;  
83897;02/02/2015;1800;30.1;;;
```

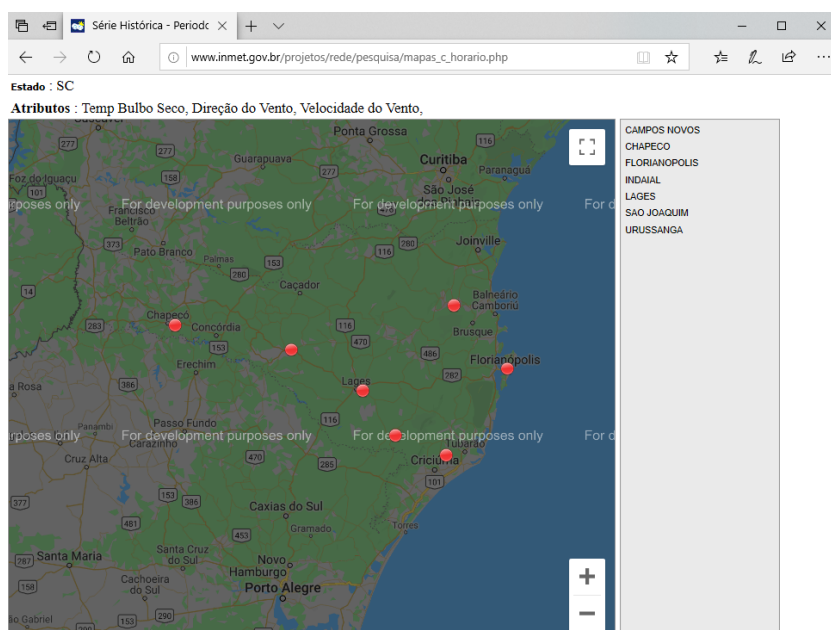
Fonte: BDMEP (2019)

Foi necessário identificar visualmente quais estações estavam dentro da área de estudo. A Figura 11 mostra a página do BDMEP onde é exibido um mapa com pontos representando as estações, o qual foi utilizado para a identificação das mesmas.

Por meio da identificação visual, foi possível conhecer quais são as estações dentro do estado de Santa Catarina. No total são 7 estações meteorológicas espalhadas pelo território, apresentadas na tabela 4, com seu código dado pelo BDMEP e seu nome.

Através da análise dos dados, foi identificado que os eventos são registrados em três horários durante o dia: à meia noite, ao meio dia e às seis da tarde. Também foi possível verificar a completude dos dados, considerando que são registrados 3 eventos diários. A Figura 5 mostra a completude dos dados obtidos no BDMEP para cada estação utilizada e na Figura 12, a sua localização no Estado de Santa Catarina.

Figura 11: Página para consulta das estações do BDMEP.



Fonte: BDMEP (2019)

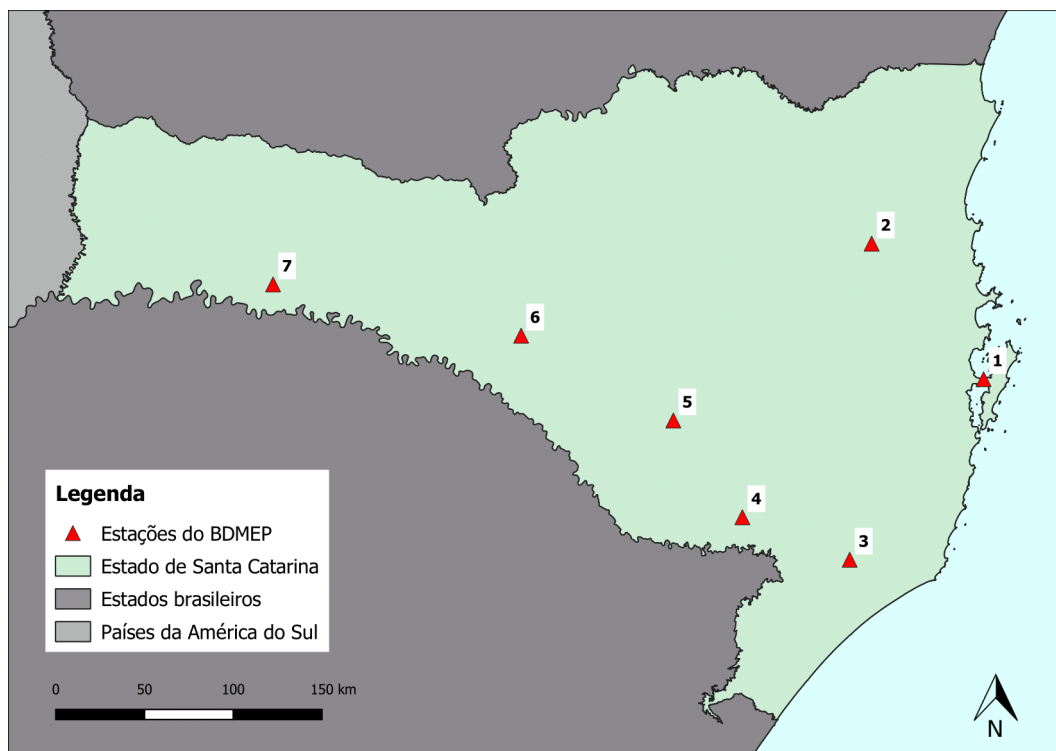
Tabela 4: Estações meteorológicas do BDMEP no Estado de Santa Catarina.

Código da estação	Nome da estação
83897	Florianópolis
83872	Indaial
83923	Urussanga
83920	São Joaquim
83891	Lages
83887	Campos Novo
83883	Chapecó

Tabela 5: Completude das estações do BDMEP consideradas no Estado de Santa Catarina

	Código da estação	Nome da estação	Anos na amostra	Total de amostras
1	83897	Florianópolis	5	3434
2	83872	Indaial	4	2146
3	83923	Urussanga	3	2086
4	83920	São Joaquim	3	2176
5	83891	Lages	3	2476
6	83887	Campos Novos	3	2503
7	83883	Chapecó	3	2106

Figura 12: Localização das estações do BDMEP no Estado de Santa Catarina.



3.2 Estudo de componentes de *software* auxiliares para o desenvolvimento

Através da busca pelos dados de vento e temperatura, foi constatada uma grande quantidade de dados, de duas fontes diferentes na internet.

Para conseguir ler os dados disponíveis nos sites, foi encontrado na internet o módulo *Requests*. O *Requests* é capaz de acessar os dados de páginas na *internet*, fazer solicitações como a autenticação de um *login*, e então receber novos dados após a autenticação. Além disso, *Requests* possui uma excelente documentação em seu site, traduzida em várias línguas e de acesso livre. (REQUESTS, 2019)

Assim, para armazenar essa grande quantidade de dados, era preciso um banco de dados com uma boa documentação para facilitar o aprendizado. Foi encontrado o SQLite, que é uma biblioteca implementada com uso de linguagem C, de código aberto e permite que a aplicação tenha um banco de dados embutido. Dessa maneira, é possível de dentro da aplicação, adicionar, atualizar ou, até mesmo, deletar dados do banco (SQLITE, 2019).

Foi preciso também, processar arquivos, lê-los e transporta-los entre pastas. O módulo *glob* consegue filtrar os arquivos desejados dentre outros em uma mesma pasta. Os arquivos provenientes das estações da NOAA são recebidos de forma compactada e, para isso, era preciso que fossem

descompactados. O módulo *gzip* pode realizar tal tarefa. Ambos são módulos de Python.

Alguns caracteres dos dados das estações meteorológicas podem não ser reconhecidos pelo Python, como acentos e pontuações, ocasionando um erro na leitura do arquivo. Então, foi utilizado o módulo *codecs* para fazer a leitura e tratar os erros com caracteres ilegíveis. Nas fontes de dados utilizadas, todos os arquivos de entrada são do tipo UTF-8 (*8-bit Unicode Transformation Format*), que é um tipo de codificação binária dada para cada caractere do arquivo.

Agora, com todos esses componentes, é preciso que os dados sejam processados, e assim realizar os cálculos de orientação da pista de pouso e decolagem e da temperatura de referência. Foi encontrado o módulo Pandas, que processa dados, realiza filtros, cálculos estatísticos e possui a capacidade de gerar gráficos. É um projeto de código aberto, exclusivo para linguagem de programação Python. Possui uma comunidade extremamente participativa, com realização de eventos para divulgação da plataforma, de melhorias e de novas funcionalidades. A documentação é muito rica, o que facilitou o desenvolvimento (PANDAS, 2019).

Foi preciso ainda processar arquivos no formato *Shapefile*, e para isso, o módulo Geopandas foi encontrado. Ele oferece uma maneira de trabalhar com os dados geográficos, e utiliza comandos semelhantes aos do módulo Pandas. Geopandas, da mesma forma que Pandas, é exclusiva para a linguagem Python. O módulo utiliza códigos de outras bibliotecas, como *shapely* e Pandas. É possível gerar novos arquivos *shapefile* com informações processadas com o auxílio das demais bibliotecas.

Outro procedimento necessário é o cálculo das superfícies interpoladas, sendo necessário um *software* de SIG para realizar essa tarefa. Através de pesquisas e orientações, foi encontrado o programa GRASS GIS, que tem a habilidade de desenvolver várias atividades com arquivos georreferenciados. Também é possível que os processos sejam automatizados na forma de *script*, utilizando Python como linguagem.

Definidos os módulos, e por meio da exclusividade de linguagem de programação imposto pelo módulo Pandas, Python foi a linguagem de programação utilizada para o desenvolvimento do trabalho.

3.3 Desenvolvimento do *software*

Agora, o procedimento de obtenção da orientação de pista apresentado na seção 2.1.2 desse trabalho, e de definição da temperatura de referência mostrado na seção 2.1.3, deve ser transformado em um processo computadorizado.

Assim, o *software* precisará dos seguintes dados de entrada:

- Estações adotadas para o estudo de cada uma das fontes de dados;
- Vento de través admissível para o aeródromo;
- Período para coleta dos dados climáticos; e
- Planilhas da ANAC com os aeródromos de referência

As estações da NOAA e do BDMEP são identificadas nas Tabelas 2 e 4, respectivamente. O *software* será capaz de fazer todo o procedimento, que é baseado em: coletar os dados climáticos das estações meteorológicas adotadas, realizar os cálculos para obter as orientações e as temperaturas de referência, fazer a interpolação dos dados e retornar a orientação de pista de pouso e decolagem e temperatura de referência.

Coleta dos dados meteorológicos

Para esta etapa, o procedimento é feito com a biblioteca *Requests*, que é capaz de interpretar páginas da internet, atribuir esses valores a variáveis, para que essas informações possam ser processadas.

O *software*, ao acessar a página da NOAA, filtra quais estações foram informadas no arquivo de entrada (*script 00*, do Apêndice B) e realiza o *download* dos arquivos padrões de cada estação desejada (*script 01*, do Apêndice C). Os arquivos dessas estações são recebidos de forma compactada. Neste mesmo passo, os arquivos são descompactados para que possam ser usados no processamento das informações de clima. Os arquivos provenientes das estações são lidos, através da biblioteca *codecs*, e inseridos no banco de dados SQLite apenas as informações contidas na Tabela 1. Então, os dados estão prontos para serem utilizados nos cálculos.

De forma semelhante, o site do BDMEP é acessado. Neste sistema é necessário realizar a autenticação do usuário, informando um *login* e senha, e realizar a solicitação de acesso. Em seguida, é possível realizar a coleta dos dados, fornecendo ao sistema quais dados estão sendo requisitados, que, neste caso, são a temperatura de bulbo seco, a direção do vento e a sua velocidade. Esta solicitação é feita para todas as estações meteorológicas do BDMEP dadas como parâmetro de entrada para o *software*. Então, os dados de temperatura, direção do vento e velocidade do vento são armazenados no banco de dados do *software* e estarão prontos para o procedimento de cálculo da orientação de pista e da temperatura de referência do aeródromo.

Cálculos para obtenção da orientação da pista de pouso e decolagem

O procedimento de cálculo é igual para as duas fontes de dados. Para a sistematização do processo, foi utilizada uma rosa dos ventos digitalizada, e representada por coordenadas em plano cartesiano. Assim, é possível que a análise dos dados seja de forma analítica.

Assim, a rosa dos ventos foi elaborada com as orientações em graus e divididas igualmente de 10 em 10 graus. As velocidades do vento foram divididas em intervalos: 0 a 3, 3 a 6, 6 a 10, 10 a 16, 16 a 21, 21 a 27, 27 a 33, 33 a 40, e maiores que 40 nós.

Na Figura 13 pode ser visto o modelo de rosa dos ventos utilizado no desenvolvimento do *software*.

Como no processo gráfico, o sistema verifica todas orientações de 0 a 180 graus, e também atribui uma porcentagem para cada orientação e intervalo de velocidade do vento. Na descrição do procedimento manual feita por Ashford, Mumayiz e Wright (2011), ao rotacionar a régua, que representa o vento de través admissível (azul na Figura 13), não é descrito em que momento a porcentagem de um célula é considerada no somatório do fator de utilização.

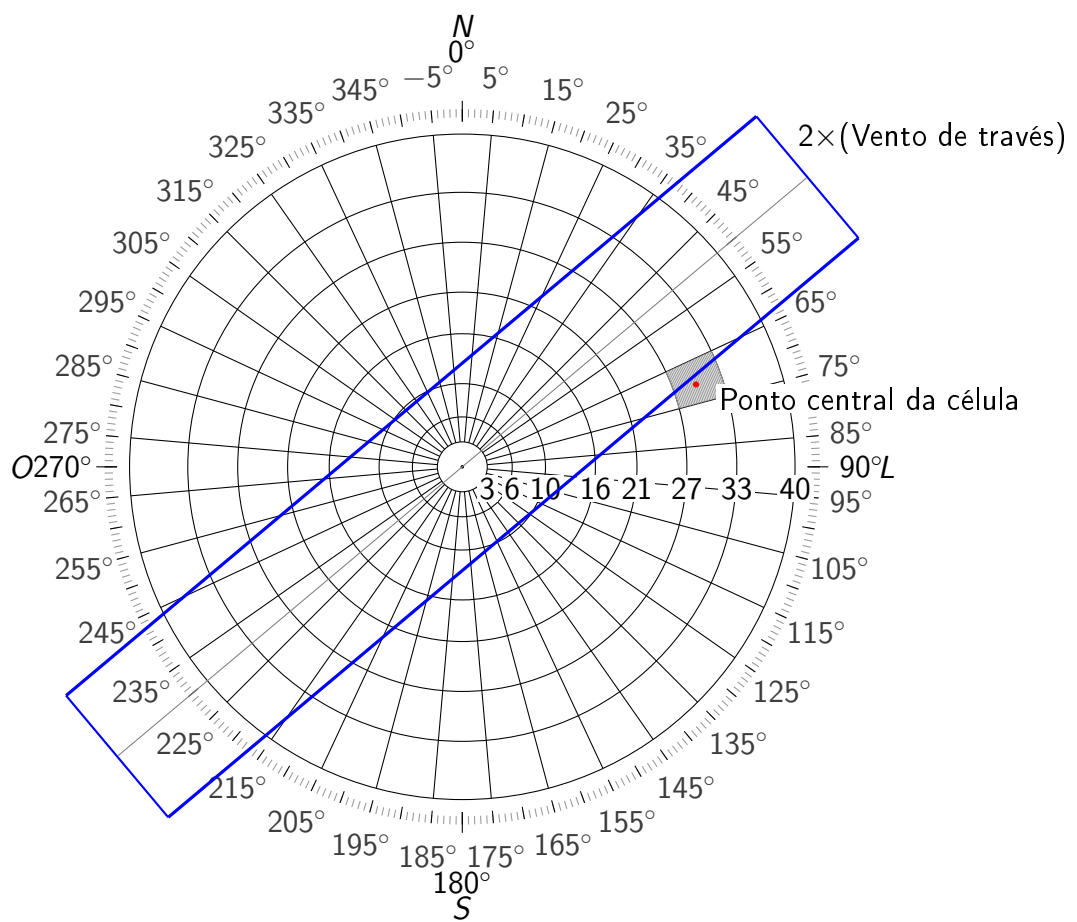
Na Figura 13, uma das células da rosa dos ventos está hachurada em cinza, dentro está seu ponto central. O presente método define que, a porcentagem será atribuída à aquela orientação, quando o ponto central estiver entre as duas retas azuis da régua.

Desta maneira, foi implementado no *software* uma tabela, com todas direções e velocidades de vento, e identificando todos os registros de vento. Então, com base no total de amostras para cada estação, foi calculada a participação de cada seção da rosa dos ventos.

Para definir a melhor orientação de pista, é utilizado o vento de través admissível para o aeroporto (representado na Figura 13). São então definidas analiticamente duas linhas equidistantes da origem da rosa dos ventos, a uma distância de igual valor ao vento de través. Estas retas são rotacionadas em torno da origem de grau em grau, somando a porcentagem da contribuição dos ventos para cada direção até 180 graus. Por fim, é considerado como a melhor orientação para a pista de pouso e decolagem, a que possuir o maior somatório das porcentagens.

Ainda, foi preciso corrigir a orientação da pista de pouso e decolagem para a declinação magnética para cada localização das estações. A base de dados da NOAA disponibiliza em seu site uma ferramenta de consulta, onde é necessário que se informe as coordenadas do lugar desejado, e ela retorna a declinação magnética. Foi implementado no *software* a correção da orientação considerando a ferramenta disponibilizada pela NOAA.

Figura 13: Modelo de rosa dos ventos utilizada.



Cálculos para obtenção da temperatura de referência

Para a determinação da temperatura de referência, com os dados previamente armazenados na base SQLite pelo processo de coleta dos dados, a forma de efetuar os cálculos é conforme apresentado na Seção 2.1.3 deste trabalho.

Foi implementado um código, para retornar a temperatura média de todos os meses da amostra, considerando todas as observações diárias. Assim, foi possível identificar qual mês teve a média mais elevada. Com essa informação, foi então necessário obter as temperaturas máximas das observações diárias e então, calcular a média dessas temperaturas.

Interpolação dos dados com o GRASS GIS

Neste ponto, a melhor orientação da pista de pouso e decolagem e a temperatura de referência para cada estação, das duas fontes, já foram calculadas. O resultado destas operações são dois arquivos, "SC_NOAA.csv" e "SC_BDMEP.csv", que possuem as informações das estações meteorológicas e as orientações e temperaturas referentes a cada estação.

Assim, para facilitar a entrada dos dados no ambiente criado pelo GRASS GIS, os arquivos citados foram convertidos para *shapefile*.

O desenvolvimento do *script* para utilizar no GRASS GIS foi feito utilizando comando para a versão 2 do Python. Até o desenvolvimento deste trabalho, o GRASS GIS não suportava *scripts* utilizando Python 3. Com isso, para o uso das funções do GRASS GIS via *script*, é necessário que sejam importadas as bibliotecas "pygrass" e "script", do pacote "grass".

Com todos arquivos prontos, foram importados para o ambiente criado no GRASS GIS. Juntamente, também foi importado o arquivo *shapefile* de Santa Catarina, de modo que fosse usado como máscara para a apresentação da imagem interpolada.

Para a interpolação, foram escolhidos os métodos IDW e uma variação do *Spline*, o Splines Regularizadas com Tensão (RST). Para os dois métodos foi utilizado 2 como parâmetro de tensão. O GRASS GIS dispõem de uma excelente documentação sobre a execução dos dois métodos, o que influenciou nesta decisão. O método de *kriging* não foi possível ser implementado, devido a incompatibilidade com o sistema operacional utilizado para desenvolver o presente trabalho.

Foram criadas oito superfícies interpoladas, uma para cada combinação: são duas fontes de dados, duas variáveis a serem analisadas e dois métodos de interpolação. Então, no *script* são utilizados os comandos "v.surf.idw" e "v.surf.rst" para gerar as superfícies contínuas das informações.

3.4 Busca e organização de dados de referência

Para validar o *software*, foi necessário encontrar dados de aeródromos existentes, de modo que fosse possível comparar os dados obtidos do *software*. A ANAC (2019) disponibiliza em seu site duas

listas, uma para aeródromos públicos e outra para privados.

Nestas listas, que são disponibilizadas no formato de planilha eletrônica, são encontrados os dados: código ICAO do aeródromo, nome, município, estado, latitude, longitude, altitude, tipos de operação, designação da cabeceira, e outros dados complementares. O formato do arquivo pode ser visto na Figura 14.

Figura 14: Arquivo padrão dos Aeródromos.

	A	B	C	D	E	F	G	H	I	J	K	L	M
	CÓDIGO ICAO	CIAD	NOME	MUNICÍPIO ATENDIDO	UF	LATITUDE	LONGITUDE	ALTITUDE	OPERAÇÃO	DESIGNAÇÃO	COMPRIMENTO	LARGURA	RES
1	SBA	PA0008	CONCEIÇÃO DO ARAGUAIA	CONCEIÇÃO DO ARAGUAIA	PA	8° 20' 55" S	49° 18' 11" W	199 m	VFR Diurno/Noturno e IFR Diurno	08/26	1800 m	30 m	PCN 47
2	SBAE	SPO010	BAURUR/AREALVA	BAURUR E AREALVA	SP	22° 9' 28" S	49° 4' 8" W	394 m	VFR Diurno/Noturno e IFR Diurno/Noturno	17/35	2010 m	45 m	PCN 42
3	SBAQ	SPO012	BARTOLOMEU DE GUSMÃO	ARARAQUARA	SP	21° 48' 18" S	48° 8' 25" W	711 m	VFR Diurno/Noturno e IFR Diurno/Noturno	17/35	1800 m	30 m	PCN 40
4	SBAR	SE0001	SANTA MARIA	ARACAJU	SE	10° 59' 7" S	37° 4' 24" W	7 m	VFR Diurno/Noturno e IFR Diurno/Noturno	11/29	2200 m	45 m	PCN 48
5	SBAT	MT0003	PILOTO OSVALDO MARQUES DIAS	ALTA FLORESTA	MT	9° 51' 59" S	56° 6' 18" W	289 m	VFR Diurno/Noturno e IFR Diurno/Noturno	04/22	2500 m	30 m	PCN 29
6	SBAU	SPO009	ESTADUAL DARIO GUARITA	ARAÇATUBA	SP	21° 8' 39" S	50° 25' 35" W	415 m	VFR Diurno/Noturno e IFR Diurno/Noturno	05/23	2120 m	35 m	PCN 26
7	SBAZ	MG0008	ROMEU ZEMA	ARAXÁ	MG	19° 33' 38" S	46° 57' 56" W	999 m	VFR Diurno/Noturno e IFR Diurno/Noturno	15/33	1900 m	30 m	PCN 26
8	SBBE	PA0001	INTERNACIONAL DE BELÉM/VAL DE CANS/ÚLIO CEZAR RIBEIRO	BELÉM	PA	1° 23' 5" S	48° 28' 44" W	17 m	VFR Diurno/Noturno e IFR Diurno/Noturno	06/24	2800 m	45 m	PCN 65
9	SBBG	RS0010	COMANDANTE GUSTAVO KRAMER	BAGÉ	RS	51° 23' 27" S	54° 6' 35" W	186 m	VFR Diurno/Noturno e IFR Diurno/Noturno	06/24	1500 m	30 m	PCN 18
10	SBBH	MG0005	PAMPULHA - CARLOS DRUMMOND DE ANDRADE	BELO HORIZONTE	MG	19° 51' 7" S	43° 57' 2" W	789 m	VFR Diurno/Noturno e IFR Diurno/Noturno	13/31	2264 m	45 m	PCN 44
11	SBBJ	PR0006	BACACHERI	CURITIBA	PR	25° 24' 12" S	49° 14' 1" W	932 m	VFR Diurno/Noturno e IFR Diurno/Noturno	18/36	1390 m	31 m	PCN 21
12	SBBP	SPO036	ESTADUAL ARTHUR SIQUEIRA	BRAGANÇA PAULISTA	SP	23° 58' 45" S	46° 32' 15" W	893 m	VFR Diurno/Noturno	16/34	1200 m	30 m	PCN 16
13	SBBR	DF0001	PRESIDENTE JUSCELINO KUBITSCHEK	BRASÍLIA	DF	15° 52' 16" S	47° 55' 7" W	1066 m	VFR Diurno/Noturno e IFR Diurno/Noturno	11L/29R	3200 m	45 m	PCN 76
14	SBBU	SPO017	BAURUR	BAURUR	SP	22° 20' 37" S	49° 3' 14" W	657 m	VFR Diurno/Noturno e IFR Diurno/Noturno	14/32	1501 m	35 m	PCN 13
15	SBBV	RR0001	ATLAS BRASIL CANTANHEDE	BOA VISTA	RR	2° 50' 29" N	60° 41' 32" W	84 m	VFR Diurno/Noturno e IFR Diurno/Noturno	08/26	2700 m	45 m	PCN 38
16	SBBW	MT0008	BARRA DO GARÇAS	BARRA DO GARÇAS	MT	15° 51' 39" S	52° 23' 22" W	350 m	VFR Diurno/Noturno e IFR Diurno/Noturno	07/25	1950 m	30 m	PCN 20
17	SBCA	PR0005	CORONEL ADALBERTO MENDES DA SILVA	CASCATEL	PR	25° 0' 8" S	53° 30' 7" W	754 m	VFR Diurno/Noturno e IFR Diurno/Noturno	15/33	1615 m	30 m	PCN 36
18	SBCB	RJ0003	CABO FRIO	CABO FRIO	RJ	22° 55' 17" S	42° 4' 18" W	6,55 m	VFR Diurno/Noturno e IFR Diurno/Noturno	10/28	2550 m	45 m	PCN 76
19	SBCD	SC0006	CARLOS ALBERTO DA COSTA NEVES	CAÇADOR	SC	26° 47' 23" S	50° 56' 23" W	1029 m	VFR Diurno/Noturno e IFR Diurno/Noturno	02/20	1625 m	30 m	PCN 25
20	SBCF	MG0001	TANCREDO NEVES	CONFINS	MG	19° 37' 28" S	43° 58' 19" W	827 m	VFR Diurno/Noturno e IFR Diurno/Noturno	16/34	3000 m	45 m	PCN 83
21	SBCG	MS0001	CAMPO GRANDE	CAMPO GRANDE	MS	20° 28' 10" S	54° 40' 13" W	559 m	VFR Diurno/Noturno e IFR Diurno/Noturno	06/24	2600 m	45 m	PCN 48
22	SBCJ	SP0004	FRANCO RIBEIRO	JUAZEIRO DO NORTE	SP	23° 4' 3" S	46° 58' 43" W	1654 m	VFR Diurno/Noturno e IFR Diurno/Noturno	11/29	1760 m	45 m	PCN 44

Fonte: ANAC (2019)

Através dos dados dos arquivos, é possível considerar a designação da pista como a real orientação da pista de pouso e decolagem. Assim, esse dado foi utilizado como variável de referência para os aeródromos existentes.

A Tabela 6 apresenta uma lista com os aeródromos utilizados como referência. Na Figura 15, pode ser vista a localização dos aeródromos considerados, com a numeração associada à Tabela 6.

Para temperatura de referência do aeródromo, foram encontradas apenas duas temperaturas de referência, através do *site*² do Departamento de Controle do Espaço Aéreo (DECEA) e do *site*³ da Rede de Meteorologia (REDEMET). Elas pode ser vistas na Tabela 7, estão indicadas relacionadas pela primeira coluna com a Figura 15.

²<<https://aisweb.decea.gov.br/?i=publicacoes&p=aip>>

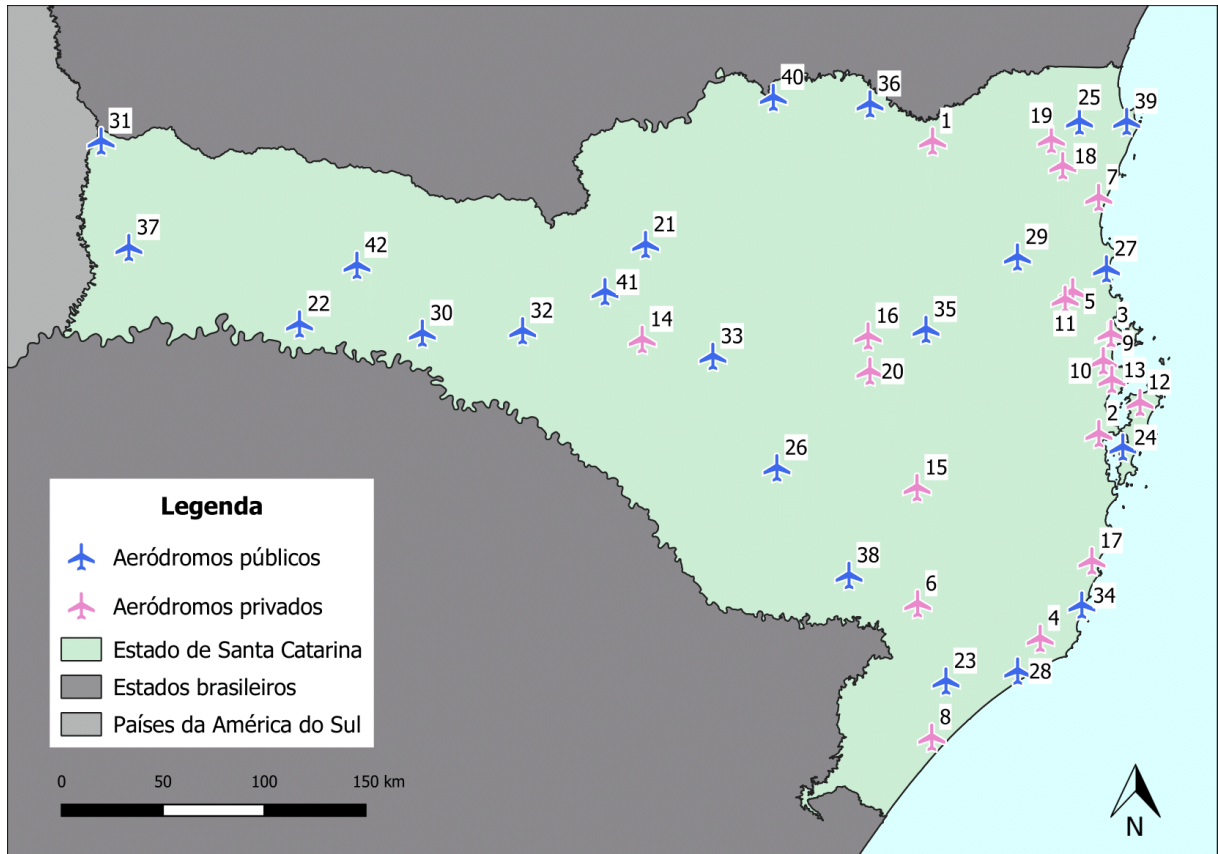
³<<https://www.redemet.aer.mil.br/?i=produtos&p=produtos-climatologicos>>

Tabela 6: Aeródromos de referência.

	Código ICAO	Nome do aeródromo	Designação da pista	Propriedade
1	SILN	Aeroclube de Rio Negrinho	11/29	Privado
2	SSKT	Aeroclube de Santa Catarina	03/21	Privado
3	SDYS	Aeroportobelo	05/23	Privado
4	SIEZ	Asa Branca	03/21	Privado
5	SIJY	Campo Comandantes	06/24	Privado
6	SINC	Campo Nuic	06/24	Privado
7	SWON	Clube de Aviação Céu Azul	16/34	Privado
8	SINN	Comandante Nelinho	07/25	Privado
9	SDEN	Costa Esmeralda	14/32	Privado
10	SIZD	DZ47	12/30	Privado
11	SWJA	Fazenda AERO-AMIL	15/33	Privado
12	SNFY	Fly Park Florianópolis	05/23	Privado
13	SJSH	Fly Ville	05/23	Privado
14	SIWE	Nelson Pizzani	08/26	Privado
15	SIJN	Pouso na Serra	01/19	Privado
16	SNZB	Rancho Sumidor	02/20	Privado
17	SSET	Sunset	06/24	Privado
18	SJVL	Vale Europeu	11/29	Privado
19	SNNV	Vila Nova	15/33	Privado
20	SSTJ	Walter Ewaldo Siegel	12/30	Privado
21	SBCD	Carlos Alberto da Costa Neves	02/20	Público
22	SBCH	Serafin Enoss Bertaso	11/29	Público
23	SSIM	Forquilha/Criciúma	09/27	Público
24	SBFL	Hercílio Luz	14/32 e 03/21	Público
25	SBJV	Lauro Carneiro de Loyola	15/33	Público
26	SBLJ	Lages	17/35	Público
27	SBNF	Ministro Victor Konder	07/25	Público
28	SBJA	Regional Sul	05/23	Público
29	SSBL	Blumenau	18/36	Público
30	SSCK	Olavo Cecco Rigon	14/32	Público
31	SSDC	Dionísio Cerqueira	01/19	Público
32	SSJA	Santa Terezinha	15/33	Público
33	SSKU	Lauro Antônio da Costa	11/29	Público
34	SSLA	Laguna	04/22	Público
35	SSLN	Helmuth Baungartem	06/24	Público
36	SSMF	Hugo Werner	03/21	Público
37	SSOE	Hélio Wasum	17/35	Público
38	SSSQ	Ismael Nunes	17/35	Público
39	SSSS	São Francisco do Sul	18/36	Público
40	SSTB	Três Barras	03/21	Público
41	SSVI	Ângelo Ponzoni	10/28	Público
42	SSXX	Municipal João Winckler	18/36	Público

Fonte: ANAC (2019)

Figura 15: *Layout* dos aeródromos de referência.



Fonte: ANAC (2019)

Tabela 7: Temperatura de referência para os aeródromos considerados como de referência.

	Código ICAO	Nome do aeródromo	Temperatura de Referência (Graus)	Fonte
24	SBFL	Hercílio Luz	29	REDEMET
27	SBNF	Ministro Victor Konder	29	DECEA

3.5 Uso do *software* para geração de resultados

Para a execução do *software* é necessário que sejam feitas configurações para que seu uso seja possível. Assim, nesta seção são apresentadas as configurações necessárias e instruções para a execução do *software*.

Pelo fato de o *software* ser escrito em Python, é necessário que o computador tenha um interpretador que seja capaz de executar *scripts* em Python. Para isso, foi usado o programa Anaconda, que possui suporte para diversas bibliotecas. Da mesma forma, para a criação das superfícies interpoladas é utilizado o GRASS GIS, ele também deve ser instalado. Os programas citados podem ser encontrados no site dos seus desenvolvedores:

- Anaconda: <<https://www.anaconda.com/distribution/>>
- GRASS GIS: <<https://grass.osgeo.org/download/>>

Também é necessária uma estrutura de pastas para o funcionamento dos *scripts*. Então, na pasta que se deseja que os *scripts* funcionem, será necessário a criação de pastas e subpastas conforme mostra a Tabela 8.

Tabela 8: Estrutura das pastas necessárias para o uso do *software*

Pasta	Descrição
aerodromos	Onde são colocados os arquivos com as informações dos aeródromos que são utilizados como referência.
aerodromos/shapefile	Subpasta da pasta "aerodromos", necessária para o processamento.
ambiente	Pasta necessária para a criação do ambiente no GRASS GIS.
pontos	Arquivo com a geometria da rosa dos ventos (encontra-se no apêndice A).
comparacao	Pasta para onde irão os arquivos das comparações com os aeroportos existentes.
dados_estacoes	Pasta necessária para o processamento.
grass_script	Pasta necessária para o processamento.
grass_script/estado	Subpasta da pasta "grass_script", onde será colocado o "shapefile" que foi baixado do site do IBGE.
grass_script/estacoes_utm	Subpasta da pasta "grass_script", pasta necessária para o processamento.
imagens	Pasta onde os arquivos <i>rasters</i> com as interpolações serão gerados.

É importante ressaltar que a grafia é *case-sensitive*, ou seja, respeitando letras minúsculas, maiúsculas e caracteres especiais.

Com o ambiente do Anaconda preparado, e a estrutura de pastas feita, é preciso agora configurar o GRASS GIS. Para isso, é necessário abrir o programa GRASS GIS e criar um *LOCATION* na pasta criada especificamente para isso.

Com todas as configurações feitas, o *software* já pode ser executado. São 6 *scripts* e devem ficar na pasta criada para o processamento. O pleno funcionamento do *software* depende da ordem de execução dos *scripts*. Para isso, é apresentada a ordem em que a execução deve ser feita.

Script nº 0 - Seleciona as estações da NOAA

O *script* "00_get_stations_noaa.py", que pode ser encontrado no Apêndice B, é responsável por fazer o filtro das estações selecionadas.

Ele deve ser executado por meio do Anaconda, utilizando seu terminal, deve-se digitar o comando: "python 00_get_stations_noaa.py".

Script nº 1 - Realiza o *download* dos dados das estações da NOAA

Esse *script* é responsável por entrar no site da NOAA e selecionar quais dados serão transferidos, através do filtro feito no *script* 00. Está presente no Apêndice C deste trabalho.

Também, executar no Anaconda: "python 01_get_station_data_noaa.py".

Script nº 2 - Processa os dados e realiza os cálculos

Esse *script* é responsável por todo o processamento dos dados e cálculos da orientação da pista de pouso e decolagem, e também, da temperatura de referência.

Pelo procedimento de obtenção dos dados do BDMEP ser diferente, esse *script* também é responsável por baixar os dados referentes às estações e processá-los.

Este *script* pode ser encontrado no Apêndice D, e deve ser executado no Anaconda: "python 02_rop.py".

Script nº 3 - Cria os *shapefiles* para uso no GRASS GIS

Nesse *script*, os cálculos das orientações de pista e da temperatura de referência já foram efetuados, e serão criados arquivos do tipo *shapefile* com os resultados.

Encontra-se no Apêndice E, e para executá-lo, é necessário inserir no Anaconda: "python 03_create_shp_stations.py".

Script nº 4 - Acessa os dados dos aeródromos existentes

Este *script* acessa os dados dos aeródromos existentes obtidos no site da ANAC (2019) e cria um *shapefile* com as informações para serem utilizadas no processamento no GRASS GIS.

O script pode ser encontrado no Apêndice F, e deve ser executado no Anaconda o comando: `"python 04_get_aeros.py"`.

Script nº 5 - Realiza a interpolação dos dados no GRASS GIS

Utiliza todos os *shapefiles* obtidos até então, realiza a interpolação dos dados e obtém a orientação da pista de pouso e decolagem e a temperatura de referência para os aeródromos existentes.

Encontra-se no Apêndice G, e desta vez, o *script* deve ser executado no GRASS GIS. Para isso, é necessário que o programa esteja aberto e com o ambiente configurado anteriormente selecionado. Então, em seu terminal de comandos, digitar: `"python 05_load_data_grass.py"`.

O resultado do processamento pode ser encontrado na pasta "aerodromos" no arquivo "aerodromos_sc.csv", que possui o ângulo das orientações de todas as pistas e as temperatura de referência para os aeródromos existentes.

3.6 Comparação e validação

Para o processo de comparação e validação dos resultados do *software*, são utilizados os arquivos que foram gerados pelo *software* e as informações obtidas das tabelas de aeródromos da ANAC.

A comparação é feita com o auxílio de um *script*, que importa os dados gerados a partir dos cálculos, realiza a comparação e apresenta os resultados.

O *script* lê os arquivos dos aeródromos a fim de encontrar a designação da pista. Este dado vem com a informação das duas cabeceiras, da forma: "XX/YY", onde em "XX" representa a cabeceira com ângulo de 1° à 180°, e em "YY" a cabeceira com ângulo de 181° à 360°.

Como fator de comparação, é utilizado o valor da primeira cabeceira multiplicado por dez para encontrar a orientação mais próxima da orientação magnética obtida no projeto do aeródromo.

Dessa forma, é obtido o ângulo observado para os aeródromos existentes. Com o valor do ângulo observado e do ângulo calculado para os aeródromos, corrigido para a declinação magnética, é feita a diferença entre eles. O *script* para comparação do ângulo da orientação da pista de pouso e decolagem pode ser encontrado no Apêndice H.

Com a temperatura de referência, serão utilizadas as duas temperaturas apresentadas na Tabela 7 como referência para os respectivos aeródromos. A comparação é feita analisando-se os dois casos, considerando-se as duas fontes de dados e os dois métodos de interpolação usados no trabalho.

4 Resultados

Nesta seção são apresentados os resultados dos cálculos efetuados pelo *software*. São expostas as orientações das pistas de pouso e decolagem e as temperaturas de referência obtidas para cada fonte de dados. Também são exibidas as orientações de pista e temperatura de referência para os aeródromos, utilizando-se as imagens obtidas da interpolação. Por fim, é feita a comparação dos resultados obtidos através do *software* com os aeródromos existentes.

4.1 Apresentação dos resultados obtidos pelo *software*

Por meio da execução do *software*, foram obtidas orientações das pistas e temperaturas de referência para cada estação meteorológica. Os resultados exibidos que são referentes às orientações de pista, consideram um vento de través admissível de 10 nós. Assim, é assumido o menor vento de través admitido.

Primeiro, para as estações da NOAA, foram obtidas as orientações de pista de pouso e decolagem e temperaturas de referência que são exibidas na Tabela 9. As orientações apresentadas estão em relação ao norte verdadeiro, então são apresentadas as declinações magnéticas a serem consideradas para cada estação meteorológica.

Através da Tabela 9, é possível identificar que apenas duas das estações obtiveram fator de utilização menor que 95%, o que indica a necessidade de uma pista auxiliar, para cobrir os ventos de través nesses aeródromos.

Também, para as estações do NOAA, foram geradas as superfícies interpoladas pelos métodos IDW e RST, tanto para a orientação da pista de pouso e decolagem, quanto para a temperatura de referência. Essas imagens são apresentadas da seguinte forma: Figura 16 - Interpolação das orientações de pista pelo método IDW; Figura 17 - Interpolação das orientações de pista pelo método RST; Figura 18 - Interpolação das temperaturas de referência pelo método IDW; e Figura 19 - Interpolação das temperaturas de referência pelo método RST.

Tabela 9: Resultados para as estações do NOAA

	Código da estação	Nome da estação	Declinação magnética (Graus)	Orientação da pista (Graus)	Fator de Utilização (%)	Temperatura de referência (°C)
1	825991	Ministro Victor Konder INTL	-19,68	72	96,31	31,13
2	827981	Lauro Carneiro de Loyola	-19,80	109	99,69	32,81
3	836727	Criciúma/Forquilha	-18,63	49	98,89	32,04
4	838830	Chapecó	-17,28	142	92,62	29,79
5	838990	Hercílio Luz	-19,48	169	98,56	31,84
6	869360	Dionísio Cerqueira	-16,91	152	97,79	28,57
7	869370	São Miguel do Oeste	-16,82	122	97,01	27,72
8	869380	Novo Horizonte	-17,38	152	98,65	24,02
9	869400	Xanxerê	-17,49	12	98,89	28,33
10	869420	Major Vieira	-18,89	119	98,67	28,82
11	869430	Caçador	-18,48	89	99,75	28,23
12	869440	Rio do Campo	-18,84	19	99,94	28,75
13	869450	Rio Negrinho	-19,37	159	100,00	28,26
14	869460	Indaial	-19,34	79	99,99	33,04
15	869470	Itapoá	-19,91	89	99,97	31,26
16	869480	Itajaí	-19,59	69	99,90	31,89
17	869550	Joaçaba	-17,93	82	99,57	29,61
18	869560	Curitibanos	-18,46	69	99,01	28,11
19	869570	Ituporanga	-18,96	109	99,94	30,53
20	869580	Florianópolis-São José	-19,46	139	99,95	29,33
21	869680	Urubici	-18,81	142	85,73	20,62
22	869690	São Joaquim	-18,50	142	99,25	24,29
23	869700	Urussanga	-18,76	159	99,98	31,44
24	869710	Araranguá	-18,51	42	98,26	30,59

Figura 16: Superfície interpolada das orientações de pista para as estações do NOAA pelo método IDW.

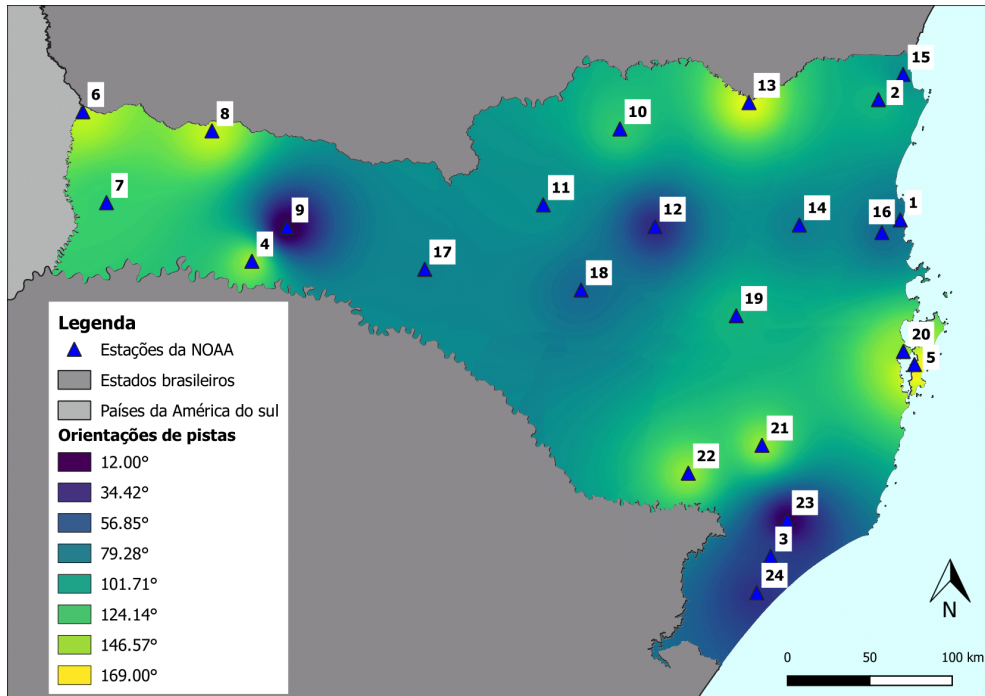


Figura 17: Superfície interpolada das orientações de pista para as estações do NOAA pelo método RST.

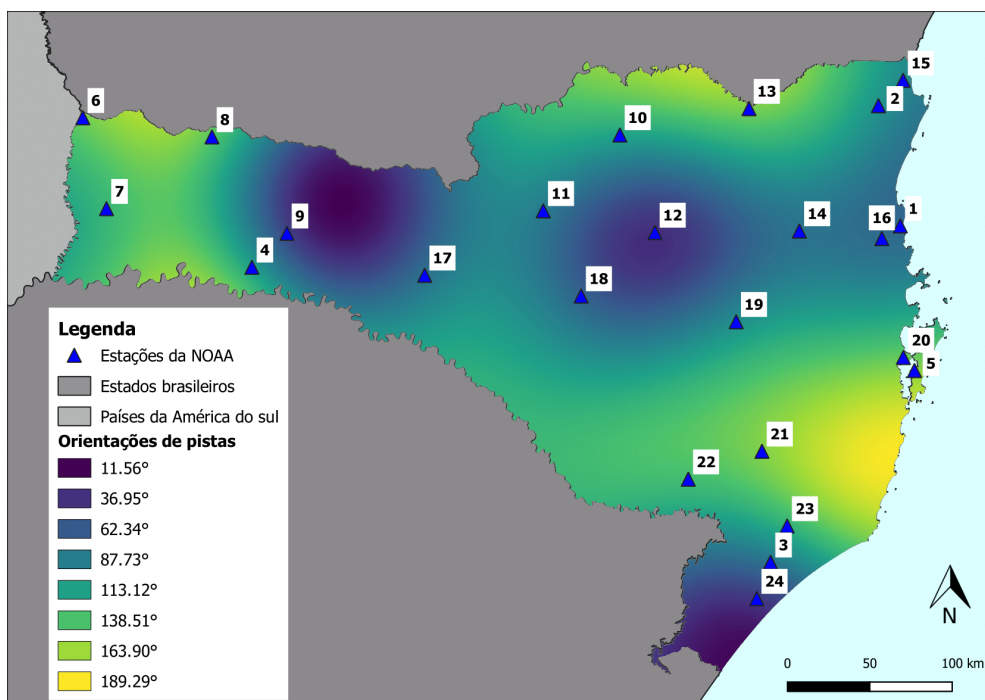


Figura 18: Superfície interpolada das temperaturas de referência para as estações do NOAA pelo método IDW.

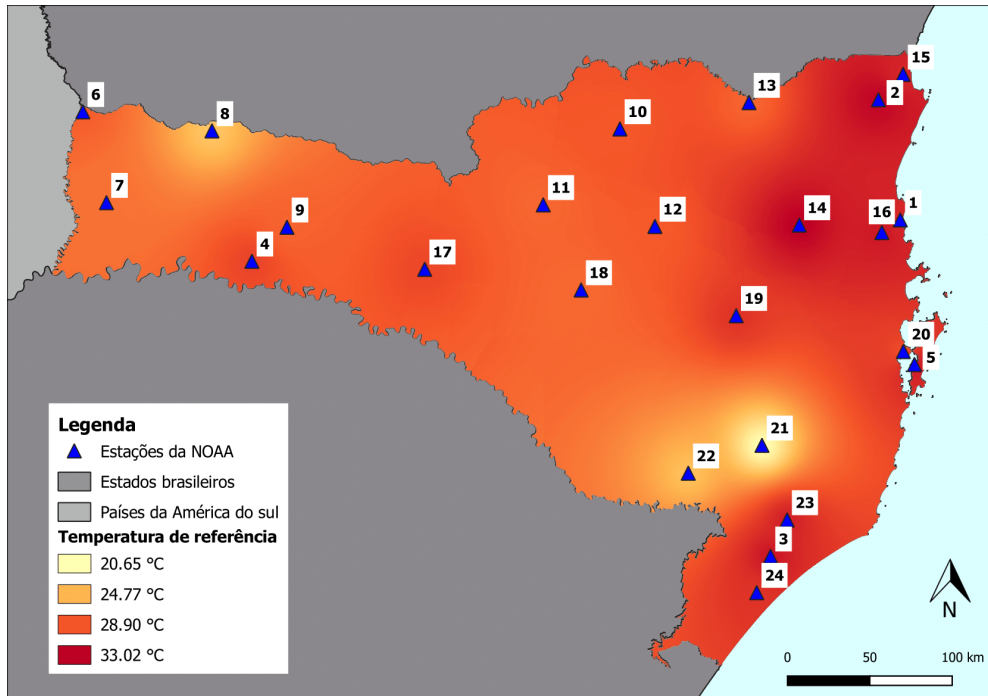
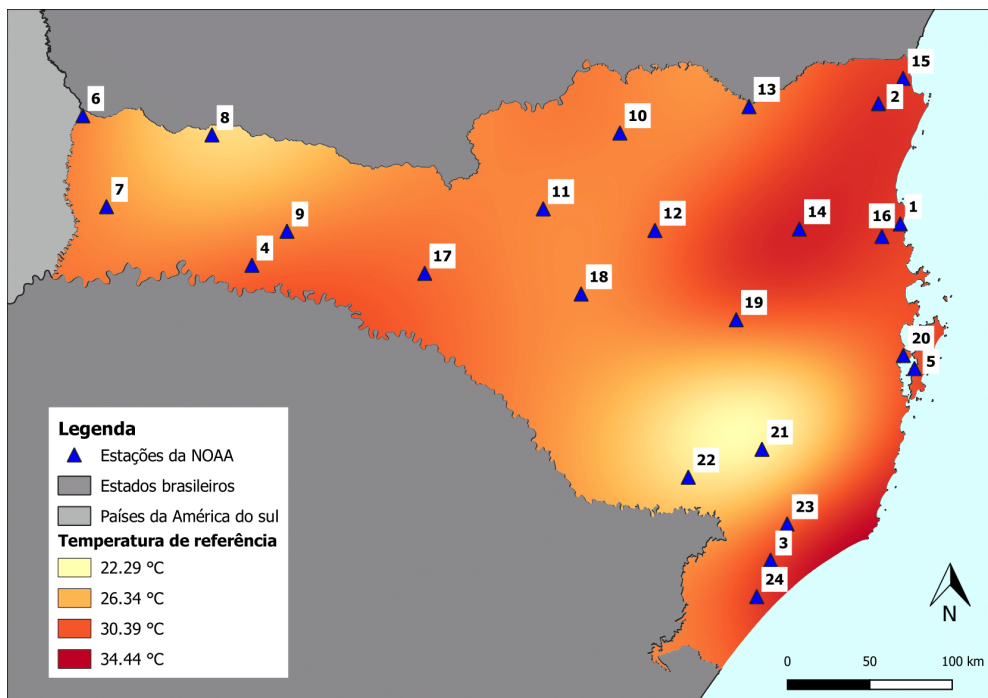


Figura 19: Superfície interpolada das temperaturas de referência para as estações do NOAA pelo método RST.



Da mesma forma que para as estações da NOAA, foram obtidos os resultados para os dados do BDMEP. Assim, as orientações de pista de pouso e decolagem e as temperaturas de referência obtidas, são exibidas na Tabela 10. Para as estações do BDMEP, o fator de utilização de todas as estações estão acima do limite mínimo de 95% recomendado para a utilização de uma pista simples.

Tabela 10: Resultados para as estações do BDMEP

	Código da estação	Nome da estação	Declinação magnética (Graus)	Orientação da pista (Graus)	Fator de Utilização (%)	Temperatura de referência (°C)
1	83897	Florianópolis	-19,50	164	99,94	29,80
2	83872	Indaial	-19,37	74	99,95	30,04
3	83923	Urussanga	-18,77	99	99,95	31,35
4	83920	São Joaquim	-18,49	164	99,13	23,98
5	83891	Lages	-18,44	99	99,88	26,41
6	83887	Campos Novos	-18,07	99	99,52	25,82
7	83883	Chapecó	-17,29	99	100,00	28,41

Foram geradas as superfícies interpoladas para as estações do BDMEP utilizando os métodos IDW e RST, para a orientação da pista de pouso e decolagem e para a temperatura de referência. Essas imagens são apresentadas da seguinte forma: Figura 20 - Interpolação das orientações de pista pelo método IDW; Figura 21 - Interpolação das orientações de pista pelo método RST; Figura 22 - Interpolação das temperaturas de referência pelo método IDW; e Figura 23 - Interpolação das temperaturas de referência pelo método RST.

Figura 20: Superfície interpolada das orientações de pista para as estações do BDMEP pelo método IDW.

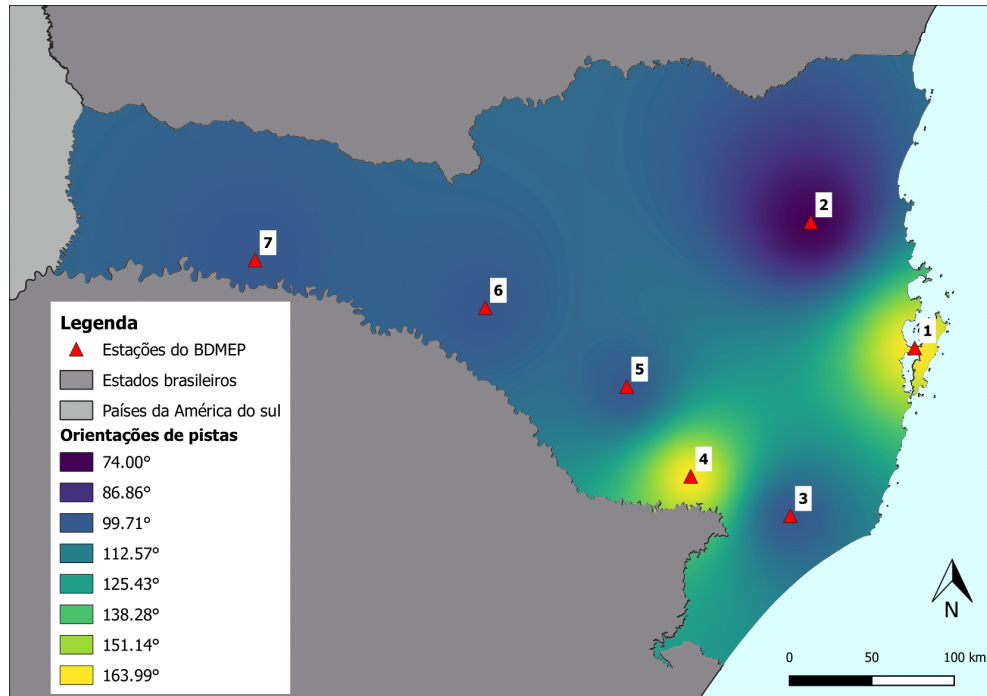


Figura 21: Superfície interpolada das orientações de pista para as estações do BDMEP pelo método RST.

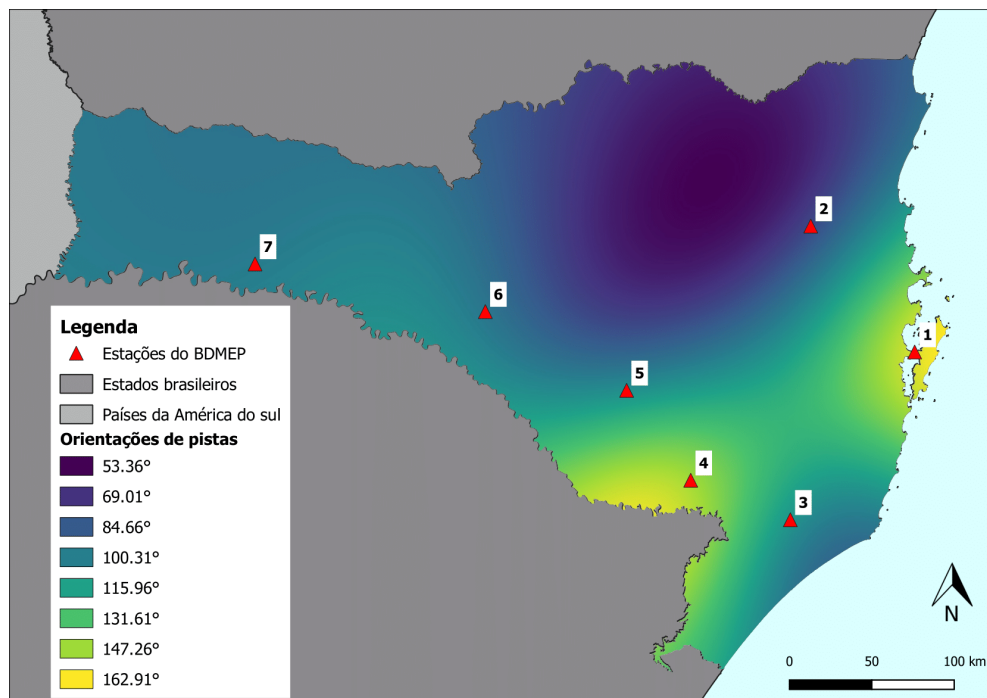


Figura 22: Superfície interpolada das temperaturas de referência para as estações do BDMEP pelo método IDW.

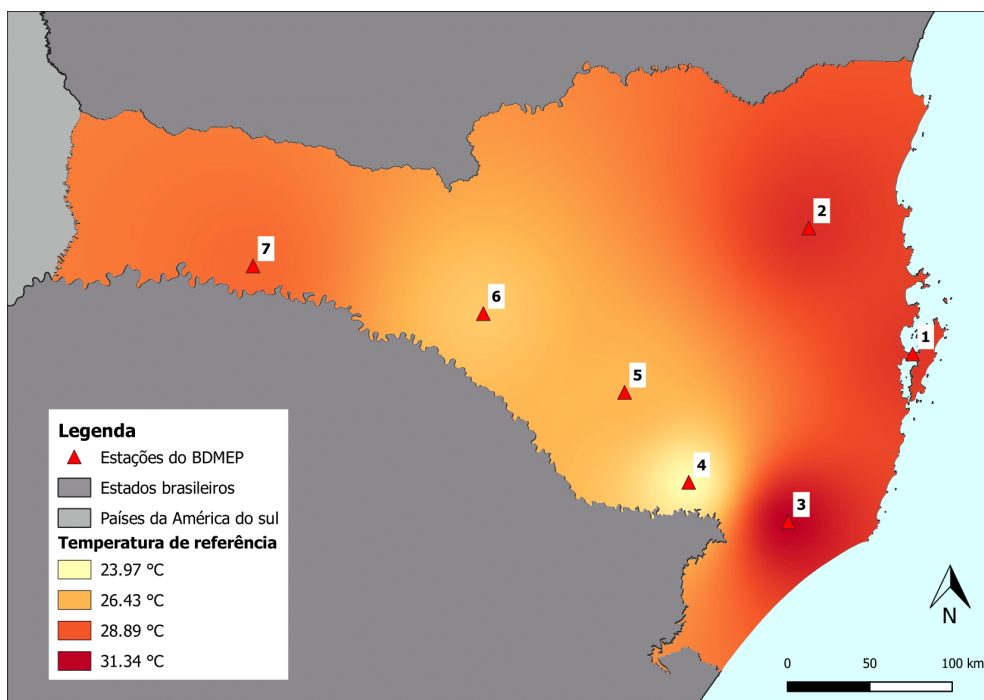
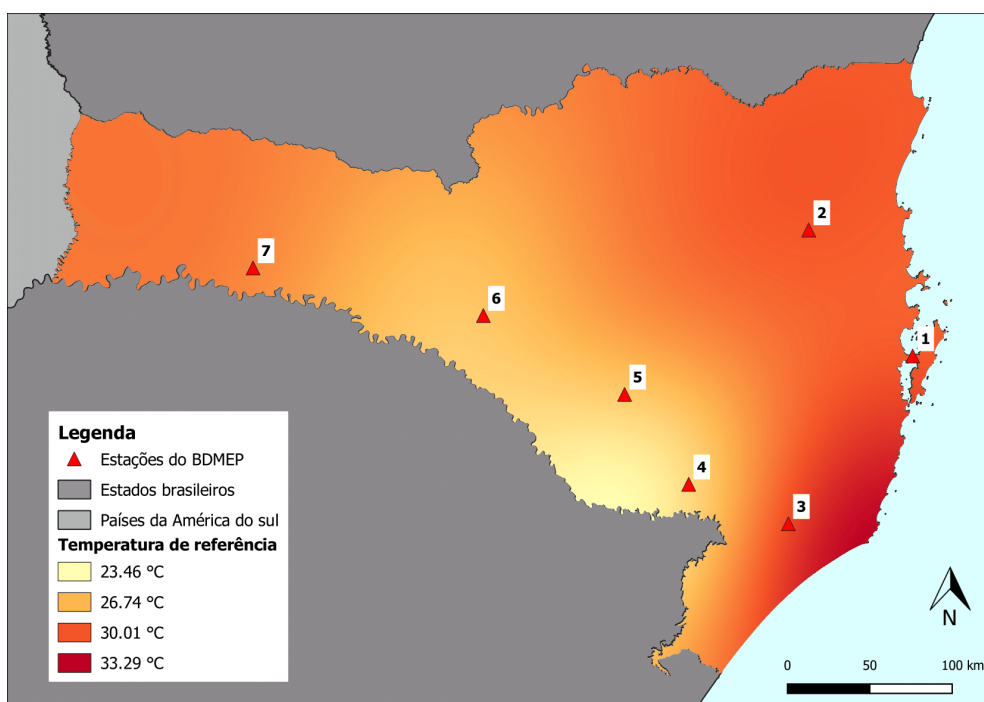


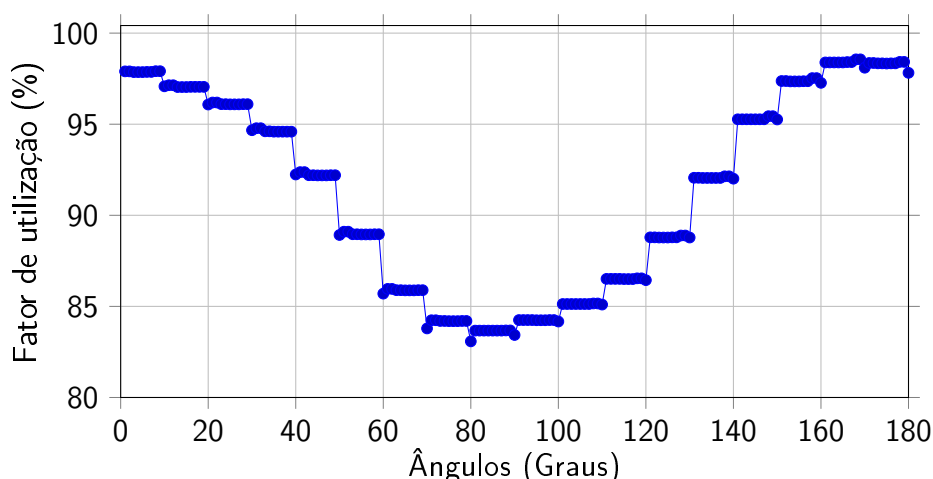
Figura 23: Superfície interpolada das temperaturas de referência para as estações do NOAA pelo método RST.



Pela análise da distribuição da porcentagem de utilização dos aeródromos, foi possível perceber uma divergência das fontes de dados. Aqui, são exemplificadas duas estações de cada fonte de dados, que ficam geograficamente próximas uma da outra. Para os dados da NOAA, serão: 838990 - Hercílio Luz; e 838830 - Chapecó. E para as estações do BDMEP, são: 83897 - Florianópolis; e 83883 - Chapecó.

Para as estações que ficam em Florianópolis, como pode ser visto nas Tabelas 3 e 5, a quantidade de amostras para as fontes de dados foi bastante diferente uma da outra. Entretanto, a distribuição dos ventos, que pode ser vista nas Figuras 24 e 25, foi semelhante uma com a outra e obtiveram um valor para a orientação de pista próximo, apesar dos dados do BDMEP mostrarem uma distribuição maior dos ventos em todas direções.

Figura 24: Fator de utilização para as orientações de pista na estação 838990 - Hercílio Luz (NOAA).



Para as estações de Chapecó, a diferença entre as completudes dos dados foi alta, e a quantidade de amostras também. Pela Tabela 9, para a estação meteorológica da NOAA, o fator de utilização é de 92,62%, o que indica a necessidade de uma pista auxiliar para atender os ventos de través. Para a estação do BDMEP, visto na Tabela 10, o fator de utilização foi de 100%, indicando que a orientação de 99 Graus do norte verdadeiro é suficiente para atender aos ventos no sítio.

Para verificar essa informação, as Figuras 26 e 27 mostram o fator de utilização para as duas estações. Para a estação da NOAA, pela distribuição do vento proveniente das amostras, percebe-se que o vento prevalece em uma direção, o que denota uma orientação de pista específica, à 142 Graus do norte verdadeiro. Para a estação do BDMEP, o cenário é distinto. A homogeneidade apresentada na Figura 27, que indica uma distribuição igual do vento em todas direções, mostra a carência de dados para o cálculo. A melhor orientação obtida para os dados do BDMEP foi de 99 Graus do norte verdadeiro.

Com base no exposto, as orientações de pistas de pouso e decolagem obtidas pelos dados da

Figura 25: Fator de utilização para as orientações de pista na estação 83897 - Florianópolis (BD-MEP).

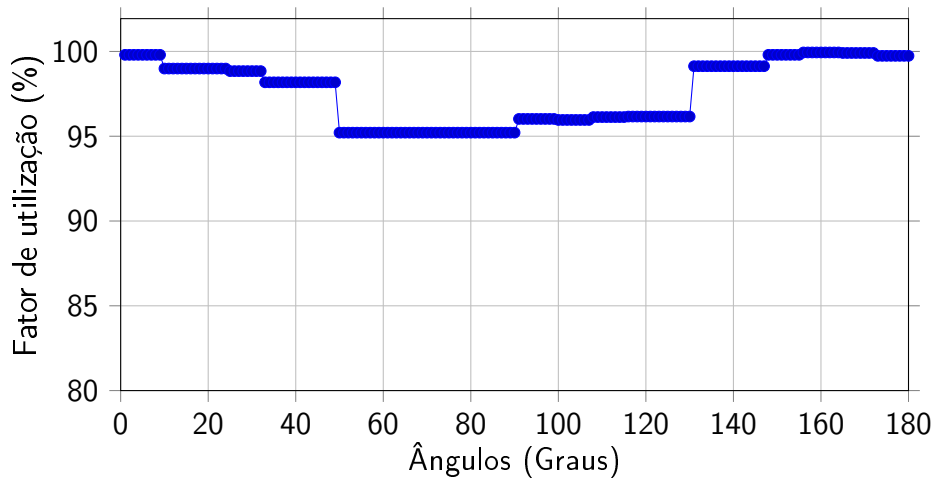


Figura 26: Fator de utilização para as orientações de pista na estação 838830 - Chapecó (NOAA).

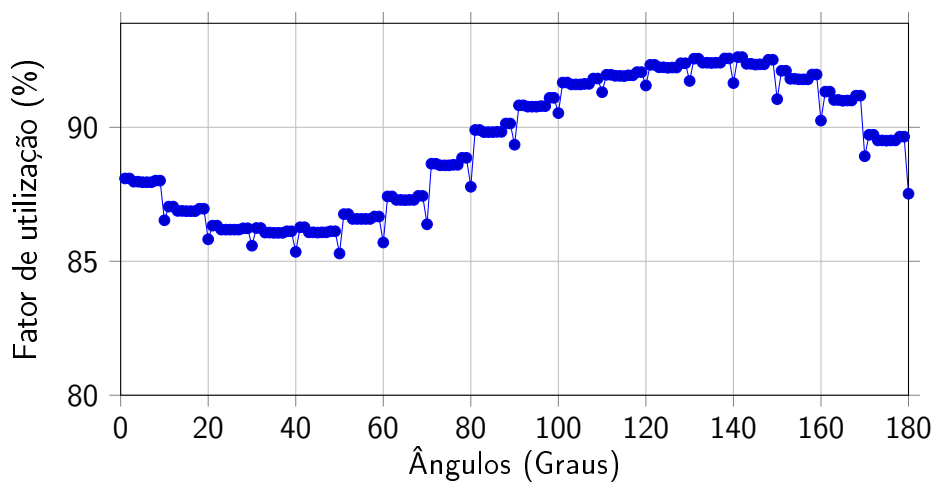
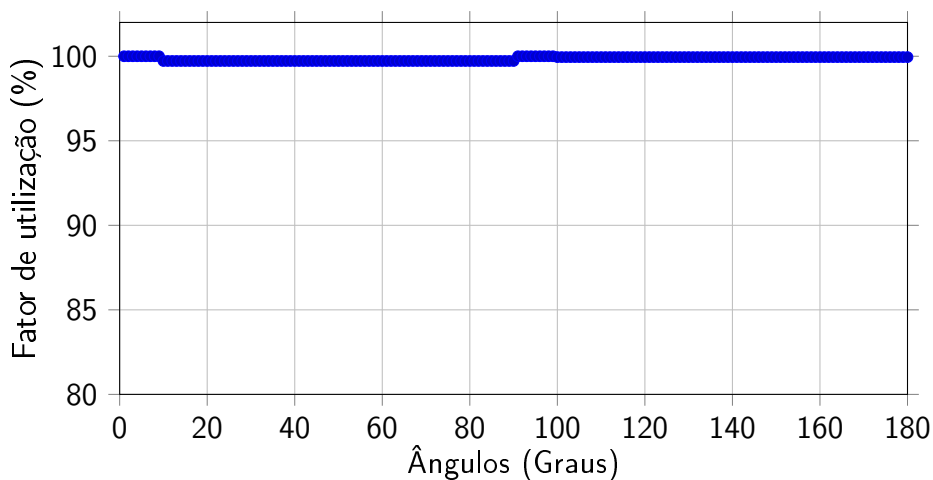


Figura 27: Fator de utilização para as orientações de pista na estação 83883 - Chapecó (BDMEP).



NOAA, consideraram uma quantidade maior de amostras do que as obtidas com os dados do BDMEP, e obtiveram os melhores resultados quanto à distribuição dos ventos para as possíveis orientações de pista.

Nas Figuras 24 e 26 é possível ver que ângulos múltiplos de 10 apresentam um valor que foge da tendência da curva, isso ocorre pelas simplificações adotadas no modelo. A atribuição da porcentagem de utilização para determinada orientação, é considerada somente quando o ponto central da célula estiver entre as linhas de limite do vento de través.

4.2 Comparação dos resultados com orientação de pistas existentes

A partir dos resultados do *software*, com as interpolações resultantes para as orientações de pista e as localizações dos aeródromos, foram obtidas as orientações das pistas de pouso e decolagem, bem como a temperatura de referência para cada aeródromo de referência.

Foram obtidas orientações de pista para as duas fontes de dados e para os dois métodos de interpolação, com intenção de compara-las com as orientações observadas das pistas, que podem ser vistas na Tabela 6. Os valores obtidos da interpolação podem ser observados na Tabela 11.

Com o objetivo de validar os resultados atingidos, foram comparadas as orientações observadas das pistas com aquelas calculadas pelo *software*. Então, através da diferença entre a orientação observada e a calculada, foram gerados gráficos de dispersão, do tipo *scatter*, que mostram a relação entre o ângulo de referência e o ângulo calculado por meio do *software*. Assim, quanto mais pontos estiverem próximos da reta $y = x$, mais próximo foi a orientação obtida pelo *software* da orientação observada na pista de pouso e decolagem.

Para os dados da NOAA, interpolados pelo método IDW, os aeródromos de referência (Figura 28) resultaram no gráfico da que pode ser visto na Figura 29. Também é possível observar que houve uma grande dispersão nos valores, com diferenças grandes. O que mostra a imprecisão causada pela interpolação dos dados de vento.

Figura 28: Superfície interpolada das orientações de pista das estações do NOAA pelo método IDW para os aeródromos de referência.

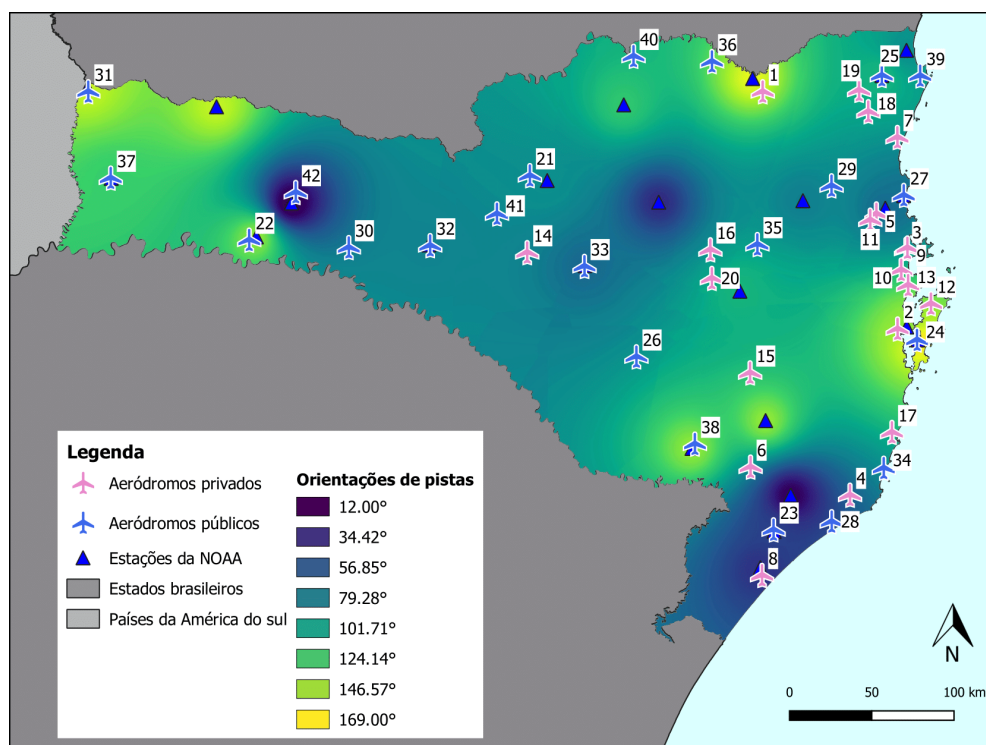
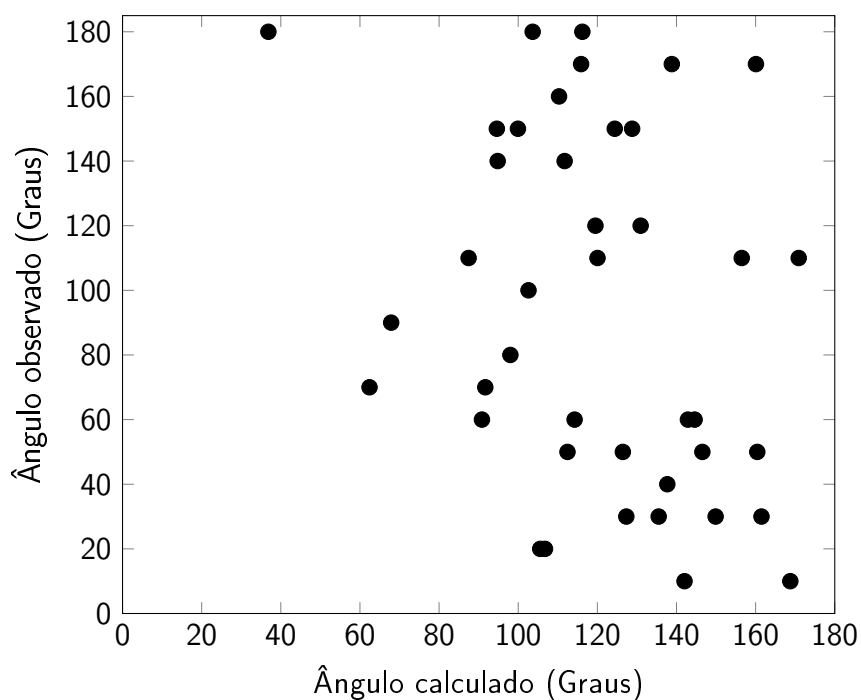


Tabela 11: Resultado das interpolações para os aeródromos de referência.

	Nome do Aeródromo	Orientações obtidas por interpolação (Graus)			
		Dados da NOAA		Dados do BDMEP	
		IDW	RST	IDW	RST
1	Aeroclube de Rio Negrinho	151,51	142,83	94,96	58,16
2	Aeroclube de Santa Catarina	142,05	148,27	161,55	151,82
3	Aeroportobelo	92,86	91,26	131,62	136,79
4	Asa Branca	116,54	153,50	109,28	96,95
5	Campo Comandantes	71,27	76,45	98,34	110,61
6	Campo Nuic	124,22	132,47	122,79	130,37
7	Clube de Aviação Céu Azul	90,57	77,85	99,16	94,34
8	Comandante Nelinho	43,92	35,90	116,48	108,14
9	Costa Esmeralda	92,13	90,63	131,32	136,86
10	DZ47	111,40	104,36	143,45	142,23
11	Fazenda AERO-AMIL	75,07	78,45	97,29	110,32
12	Fly Park Florianópolis	140,84	132,15	160,56	159,54
13	Fly Ville	127,00	115,95	153,74	149,07
14	Nelson Pizzani	79,74	91,35	101,19	85,14
15	Pouso na Serra	123,18	146,75	125,28	121,25
16	Rancho Sumidor	86,62	60,37	103,85	67,15
17	Sunset	125,36	188,85	128,65	124,97
18	Vale Europeu	100,34	97,19	95,25	80,27
19	Vila Nova	104,70	110,34	96,63	76,04
20	Walter Ewaldo Siegel	100,63	79,82	108,14	77,22
21	Carlos Alberto da Costa Neves	88,34	92,06	103,87	80,77
22	Serafin Enoss Bertaso	139,24	122,00	99,03	99,48
23	Forquilha/Criciúma	49,26	81,71	107,80	109,85
24	Hercílio Luz	168,90	157,40	162,45	157,30
25	Lauro Carneiro de Loyola	109,00	102,98	100,52	81,53
26	Lages	97,41	119,08	99,49	105,92
27	Ministro Victor Konder	72,00	71,61	106,05	114,80
28	Regional Sul	107,59	120,75	106,80	92,00
29	Blumenau	84,19	80,79	76,62	81,96
30	Olavo Cecco Rigon	77,19	54,82	101,50	103,84
31	Dionísio Cerqueira	151,87	149,22	103,95	97,41
32	Santa Terezinha	82,00	79,03	101,36	101,81
33	Lauro Antônio da Costa	69,00	69,30	104,91	74,52
34	Laguna	118,55	179,42	119,23	108,46
35	Helmuth Baungartem	95,14	78,12	94,66	75,07
36	Hugo Werner	130,63	162,42	100,77	60,65
37	Hélio Wasum	122,00	129,50	102,52	97,77
38	Ismael Nunes	141,56	136,45	163,17	145,93
39	São Francisco do Sul	96,29	85,14	104,88	91,67
40	Três Barras	108,32	154,18	104,32	67,62
41	Ângelo Ponzoni	84,36	91,96	101,80	89,31
42	Municipal João Winckler	19,34	40,54	100,04	98,43

Figura 29: Gráfico da relação entre as orientações observadas e as calculadas pelo *software* para os dados da NOAA pelo método IDW.



Nas interpolações dos dados da NOAA, pelo método RST, para os aeródromos de referência (Figura 30), o resultado não foi diferente. A diferença entre os ângulos observados e os obtidos no *software*, pode ser percebida pela análise do gráfico da Figura 31. O que nos mostra a divergência nos valores obtidos em relação aos da pista real.

Figura 30: Superfície interpolada das orientações de pista das estações do NOAA pelo método RST para os aeródromos de referência.

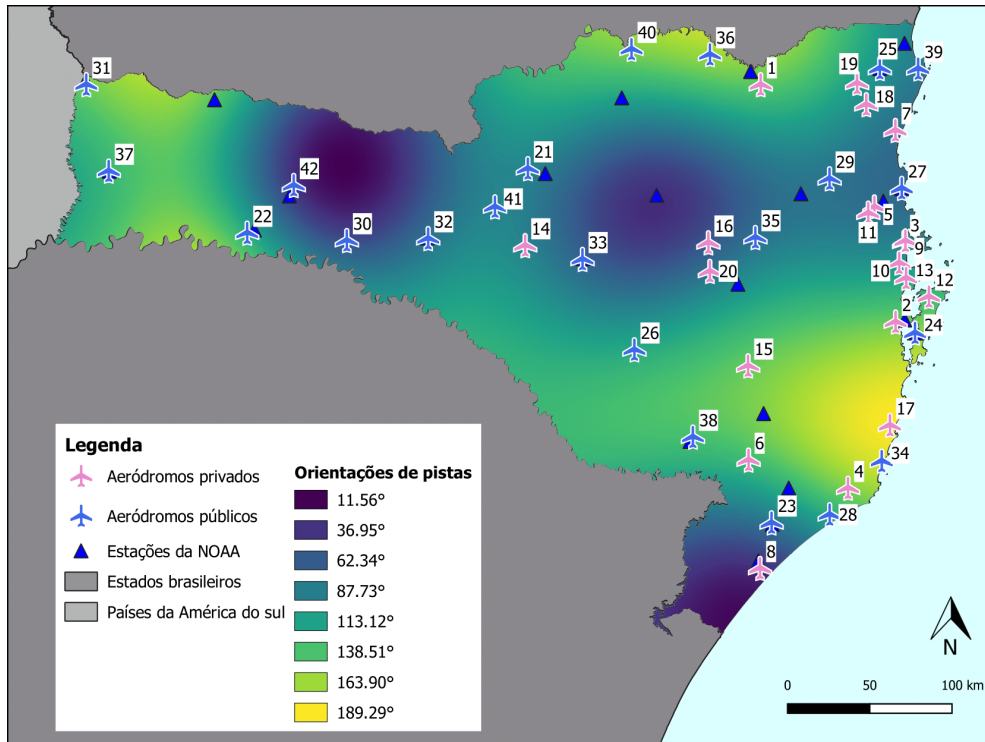
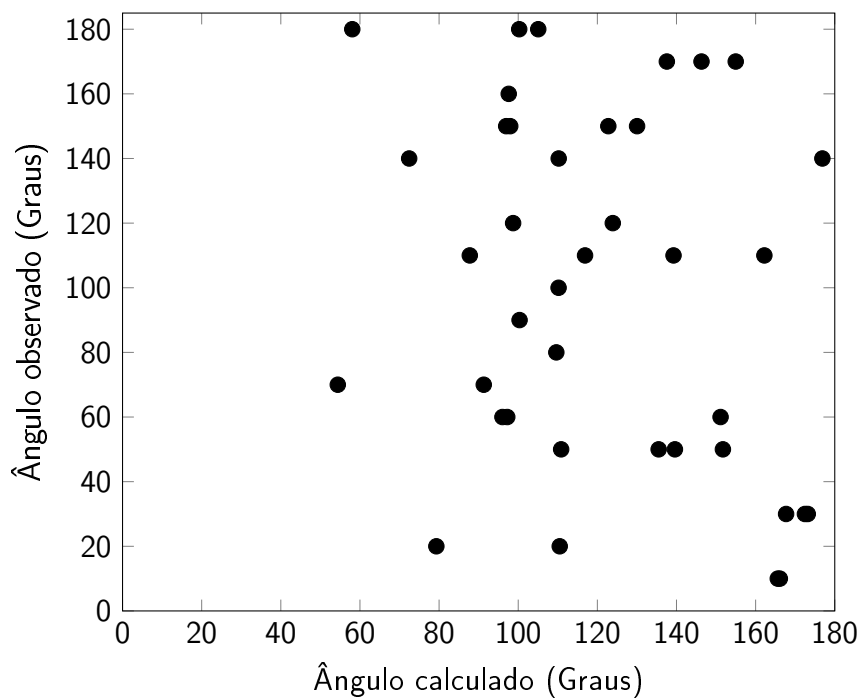


Figura 31: Gráfico da relação entre as orientações observadas e as calculadas pelo *software* para os dados da NOAA pelo método RST.

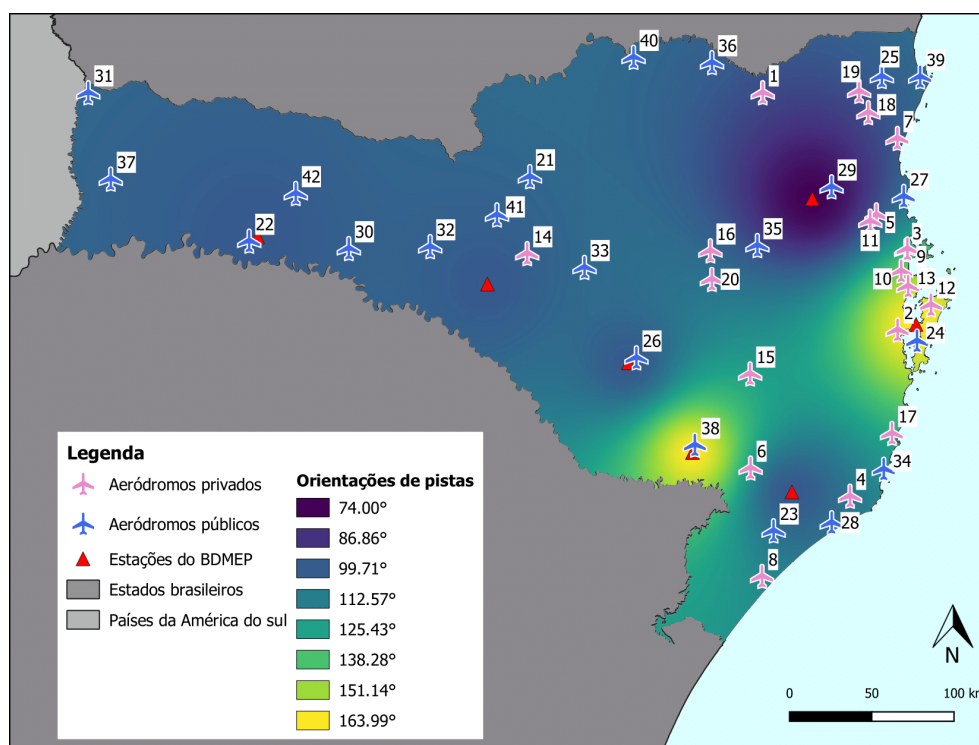


Foram obtidos resultados para as estações meteorológicas do BDMEP. Como foi visto na Seção 4.1, os resultados do BDMEP apresentaram comportamento semelhantes aos da NOAA. Assim, os valores obtidos para as orientações são comparados com os observados dos aeródromos.

Com a superfície interpolada, os valores para as orientações dos aeródromos foram obtidos, da mesma forma que o processo das estações da NOAA. No primeiro método, a interpolação IDW deu origem à superfície que pode ser vista na Figura 32, e os valores das orientações obtidas estão na Tabela 11.

Na Figura 33, podemos ver o gráfico que relaciona o ângulo de referência e o ângulo calculado com uso do *software* pela interpolação IDW. A grande dispersão mostra que os valores não são representativos para a realidade.

Figura 32: Superfície interpolada das orientações de pista das estações do BDMEP pelo método IDW para os aeródromos de referência.



Para o método RST também foram calculadas as orientações e a Figura 34 mostra a superfície interpolada. Da mesma forma, os resultados são divergentes dos observados nos aeródromos de referência. A Figura 35 mostra o gráfico da dispersão do resultado, entre os ângulos de referência e os calculados.

Figura 33: Gráfico da relação entre as orientações observadas e as calculadas pelo *software* para os dados da BDMEP pelo método IDW.

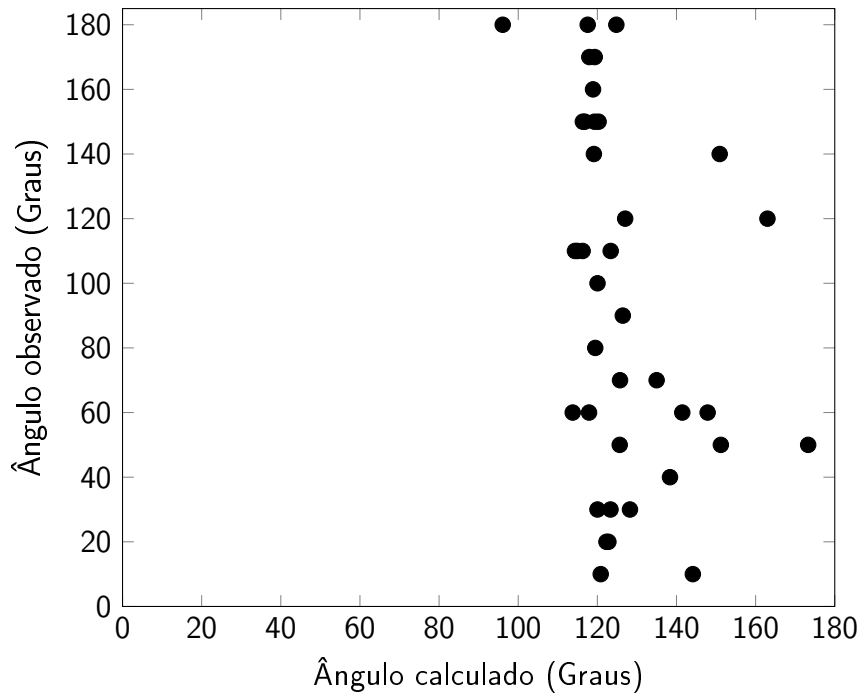


Figura 34: Superfície interpolada das orientações de pista das estações do BDMEP pelo método RST para os aeródromos de referência.

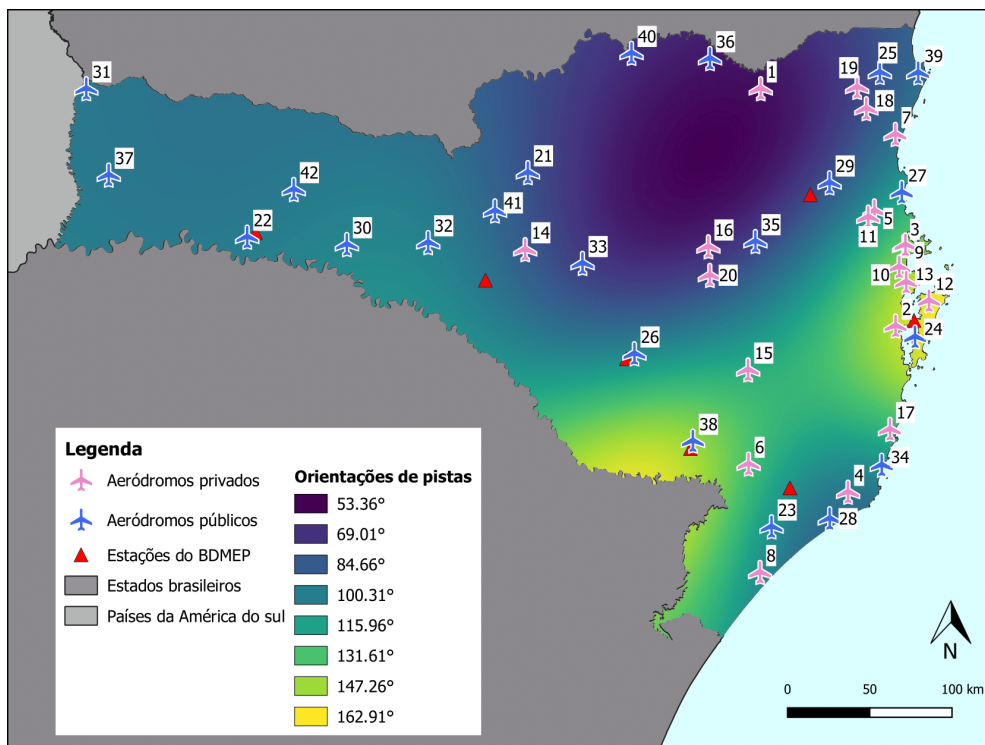
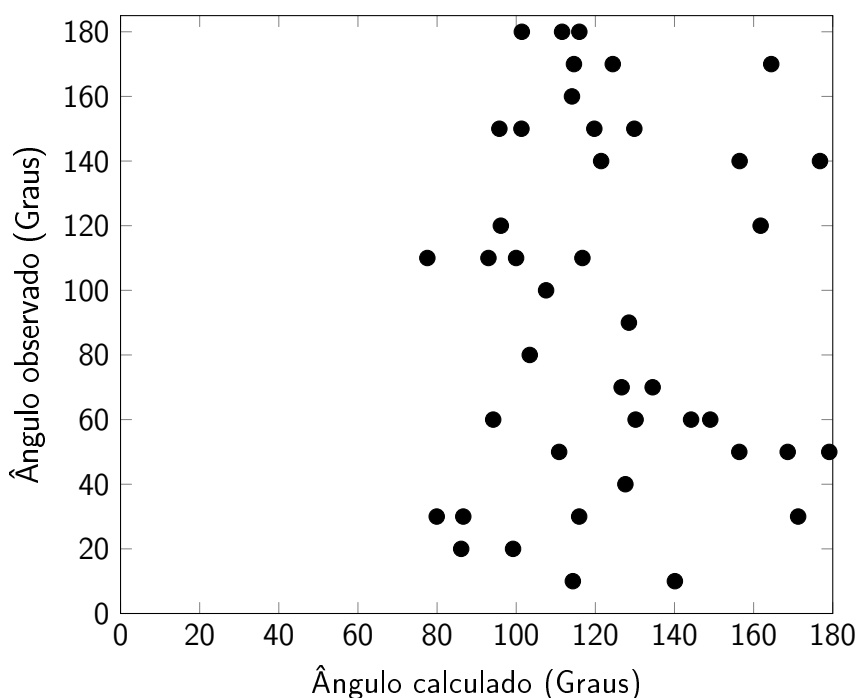


Figura 35: Gráfico da relação entre as orientações observadas e as calculadas pelo *software* para os dados da BDMEP pelo método RST.



Como fator de comparação, é observada a orientação de duas pistas de pouso e decolagem, para aeródromos distintos em relação à sua proximidade com uma estação de aquisição de dados. Um deles, o aeródromo se localiza muito próximo à uma estação meteorológica. O outro está distante das estações.

A comparação é feita com os aeroportos Hercílio Luz, em Florianópolis, e Olavo Cecco Rigon, em Concórdia, sendo que o aeródromo de Florianópolis fica muito próximo a uma estação, e o de Concórdia distante.

Na Tabela 12, é possível ver as informações do aeródromo de Florianópolis. Sua pista principal possui designação 14/32, e para a secundária 03/21, então, os ângulos observados para as pistas foram de 140 e 30 graus em relação ao norte magnético. Para as duas fontes de dados e os dois métodos de interpolação, foram calculadas as orientações de pista para o norte magnético, atribuindo a declinação magnética de -19,47 graus. Ambas fontes de dados e métodos de interpolação obtiveram ângulos com grande diferença dos observados. Entretanto, para a fonte de dados NOAA, com interpolação pelo método IDW a diferença com a pista secundária de 21,63 graus.

O aeródromo de Concórdia, que possui também sua pista com designação 14/32, também apresentou resultados diferentes das orientações observadas. Na Tabela 13, podem ser vistas as informações referentes ao aeródromo. O ângulo observado foi de 140 graus em relação ao norte magnético. Para as duas fontes de dados e os dois métodos de interpolação, os resultados divergiram

Tabela 12: Informações e cálculos para o Aeroporto Internacional Hercílio Luz.

Aeroporto Internacional Hercílio Luz	
Código ICAO	SBFL
Nome	Hercílio Luz
Município	Florianópolis
UF	SC
Designação	14/32 e 03/21
Propriedade	público
Declinação magnética	-19,47
Ângulo observado	140
Ângulo calculado dados NOAA por IDW	188,37
Ângulo calculado dados NOAA por RST	176,87
Ângulo calculado dados BDMEP por IDW	181,92
Ângulo calculado dados BDMEP por RST	176,77

do observado. A fonte de dados e o método que mais se aproximaram foi o BDMEP com interpolação RST, que atingiram um ângulo de 121,46 graus, com 18,54 graus de variação em relação ao ângulo observado.

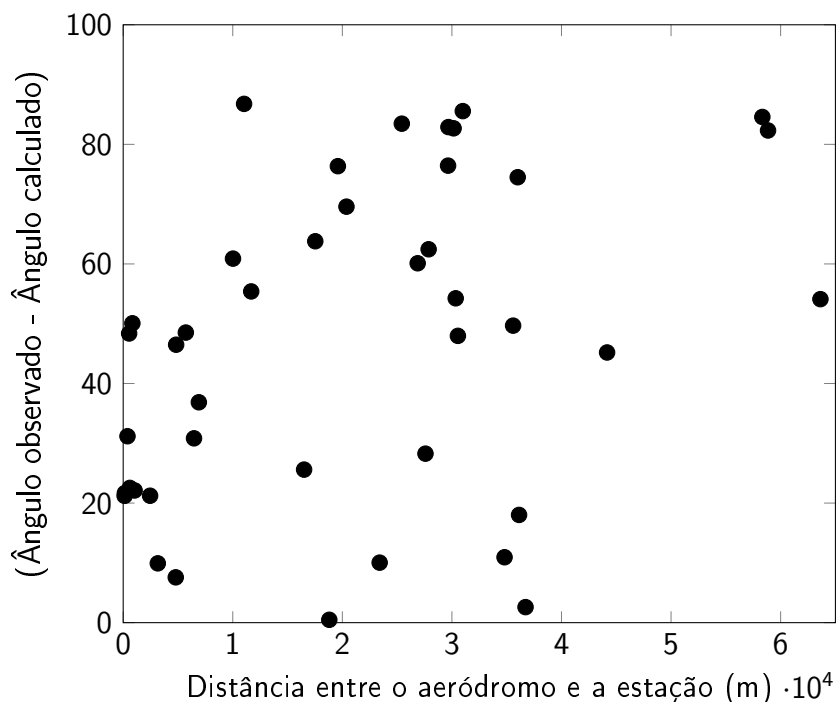
Tabela 13: Informações e cálculos para o Aeroporto Olavo Cecco Rigon.

Aeroporto Olavo Cecco Rigon	
Código ICAO	SACK
Nome	Olavo Cecco Rigon
Município	Concórdia
UF	SC
Designação	14/32
Propriedade	público
Declinação magnética	-17,62
Ângulo observado	140
Ângulo calculado NOAA IDW	94,81
Ângulo calculado NOAA RST	72,44
Ângulo calculado BDMEP IDW	119,12
Ângulo calculado BDMEP RST	121,46

Esta comparação foi feita para todos os aeródromos, relacionando a diferença entre o ângulos calculados com uso do *software* e o ângulo observado, e comparada com a distância até a estação meteorológica mais próxima.

Para os dados no NOAA, com interpolação IDW, foi gerada a Figura 36. Mostrou grande dispersão, e não apresentou uma relação entre a diferença dos ângulos com a distância de uma estação meteorológica.

Figura 36: Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do NOAA mais próxima, pelo método IDW.



Para as mesmas estações, com o método de interpolação RST, foi obtido o gráfico da Figura 37. Da mesma forma, não apresentou uma relação com a distância.

Para as estações do BDMEP, para interpolação IDW, deu origem a Figura 38, que também apresentou uma grande dispersão das diferenças dos ângulos com as distâncias até uma estação meteorológica.

Para interpolação RST, deu origem a Figura 39, que da mesma forma, apresentou uma dispersão da diferença dos ângulos com a distância até uma estação mais próxima.

Os resultados obtidos para ambos métodos de interpolação e fontes de dados, apresentaram grande diferença em relação as orientações de pista observadas. Os métodos se mostraram imprecisos para a determinação da orientação de pista de pouso e decolagem.

Figura 37: Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do NOAA mais próxima, pelo método RST.

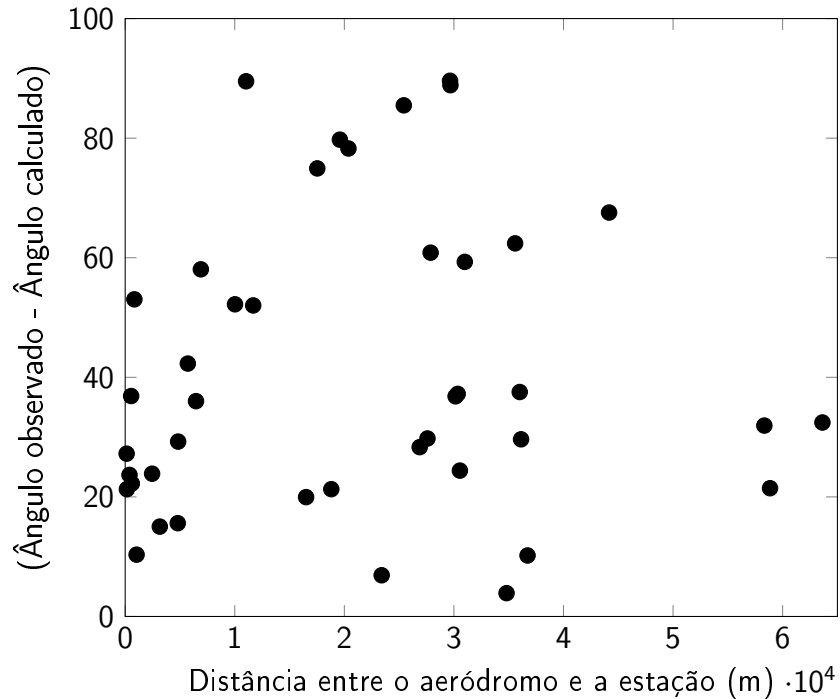


Figura 38: Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do BDMEP mais próxima, pelo método IDW.

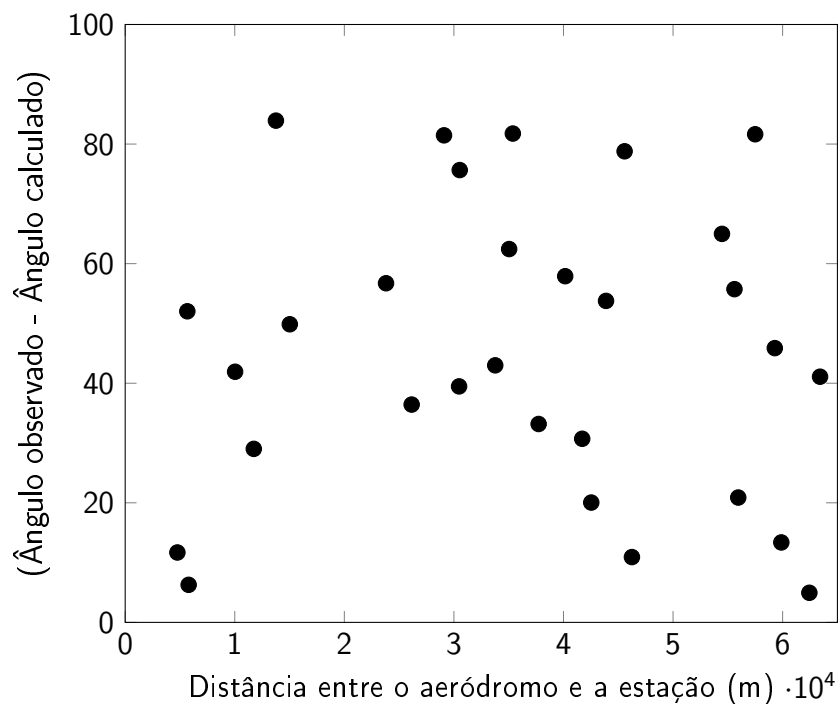
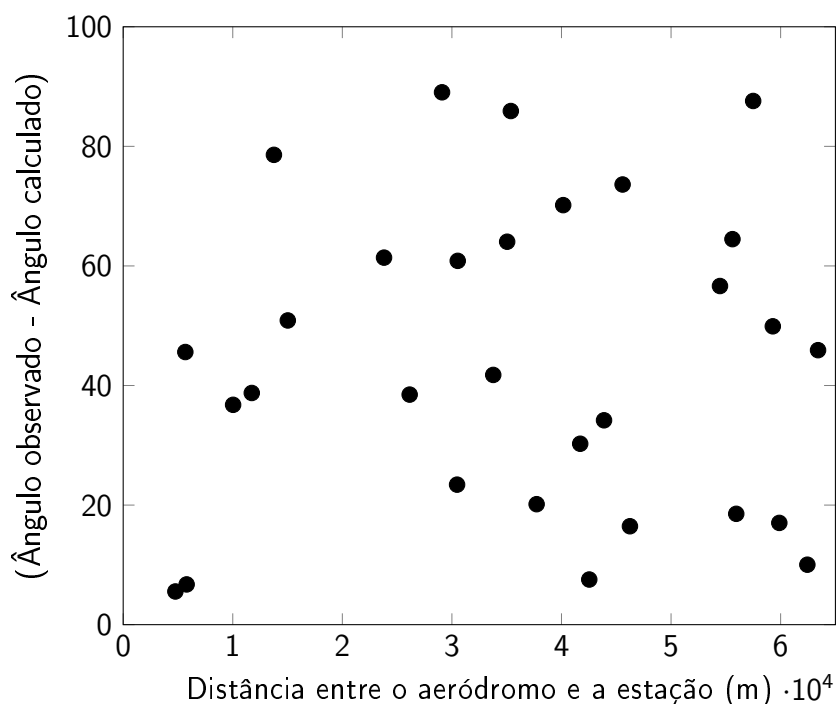


Figura 39: Gráfico da relação entre a diferença do ângulo de referência e o observado e a distância até a estação meteorológica do BDMEP mais próxima, pelo método RST.



4.3 Comparação dos resultados obtidos para temperatura de referência

As temperaturas de referência foram calculadas para os aeródromos definidos como de referência e feita a comparação com as temperaturas de referência obtidas nos dados do DECEA e da REDEMET, para dois aeródromos.

Foram determinadas as temperaturas de referência para os aeródromos utilizando os mesmos métodos de interpolação que para as orientações de pista. Através das interpolações das temperaturas se obteve a Tabela 14 e as superfícies para cada um dos métodos e fonte de dados, que são mostradas nas Figuras 40, 41, 42 e 43.

Também, por meio da Tabela 14, é possível observar a variação da temperatura entre as diferentes fontes de dados e os métodos de interpolação.

Tabela 14: Resultado das interpolações para temperatura de referência nos aeródromos.

	Nome do Aeródromo	Temperaturas de referência (°C)			
		Dados da NOAA		Dados do BDMEP	
		IDW	RST	IDW	RST
1	Aeroclube de Rio Negrinho	28,54	29,31	29,04	29,94
2	Aeroclube de Santa Catarina	29,69	29,98	29,76	30,02
3	Aeroportobelo	31,15	30,97	29,51	29,60
4	Asa Branca	29,87	33,04	30,22	32,75
5	Campo Comandantes	31,78	31,84	29,62	29,76
6	Campo Nuic	27,21	25,58	28,42	27,98
7	Clube de Aviação Céu Azul	31,48	32,17	29,40	29,92
8	Comandante Nelinho	30,60	31,58	28,99	30,02
9	Costa Esmeralda	31,17	30,98	29,51	29,60
10	DZ47	30,78	30,75	29,58	29,62
11	Fazenda AERO-AMIL	31,66	31,86	29,64	29,76
12	Fly Park Florianópolis	30,45	30,97	29,74	29,75
13	Fly Ville	30,48	30,64	29,67	29,66
14	Nelson Pizzani	28,41	28,40	26,13	26,70
15	Pouso na Serra	25,82	23,11	27,29	27,89
16	Rancho Sumidor	29,58	30,89	28,11	29,14
17	Sunset	29,47	31,38	29,36	32,07
18	Vale Europeu	31,90	32,26	29,38	30,03
19	Vila Nova	32,17	31,95	29,27	30,03
20	Walter Ewaldo Siegel	29,90	29,95	27,74	28,75
21	Carlos Alberto da Costa Neves	28,28	28,38	26,77	27,38
22	Serafin Enoss Bertaso	29,73	29,52	28,40	28,39
23	Forquilha/Criciúma	32,03	31,24	30,22	30,11
24	Hercílio Luz	31,83	30,70	29,77	30,18
25	Lauro Carneiro de Loyola	32,81	31,98	29,16	29,98
26	Lages	27,75	26,23	26,41	26,05
27	Ministro Victor Konder	31,13	31,57	29,49	29,73
28	Regional Sul	30,47	33,72	30,41	32,57
29	Blumenau	32,04	32,67	29,96	29,97
30	Olavo Cecco Rigon	28,73	30,08	27,63	27,10
31	Dionísio Cerqueira	28,56	28,28	27,89	28,97
32	Santa Terezinha	29,61	29,53	26,35	26,28
33	Lauro Antônio da Costa	28,11	28,10	26,61	27,40
34	Laguna	29,42	32,84	29,57	32,80
35	Helmuth Baungartem	30,62	32,06	29,08	29,52
36	Hugo Werner	29,03	27,88	28,52	29,61
37	Hélio Wasum	27,72	27,84	28,00	28,98
38	Ismael Nunes	24,34	23,77	24,04	25,28
39	São Francisco do Sul	31,67	31,93	29,10	29,90
40	Três Barras	29,11	28,42	27,95	29,03
41	Ângelo Ponzoni	28,65	28,62	26,31	26,71
42	Municipal João Winckler	28,35	28,01	28,20	28,13

Figura 40: Superfície interpolada das temperaturas de referência das estações da NOAA pelo método IDW para os aeródromos de referência.

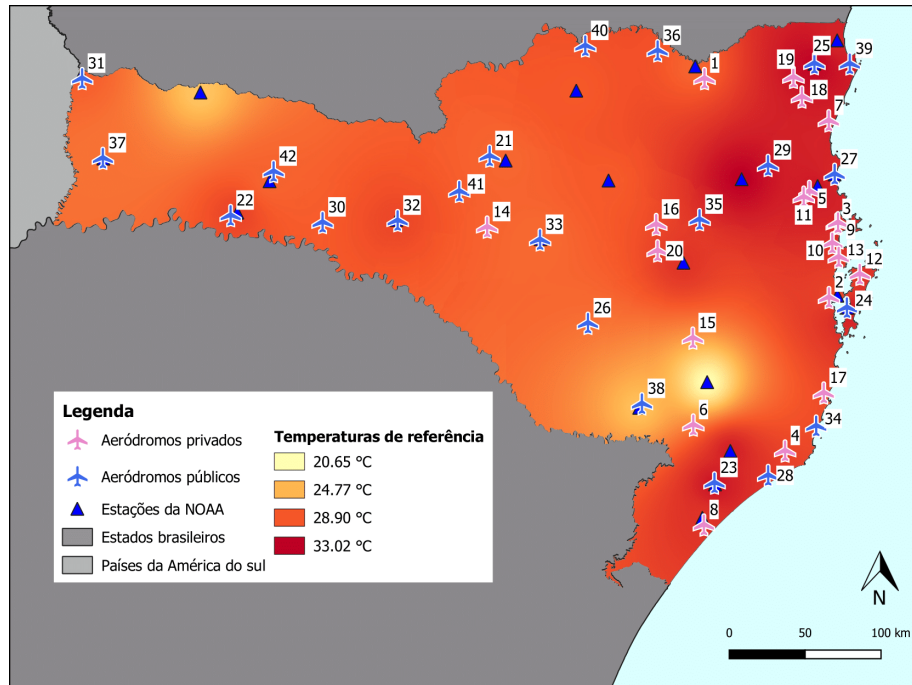


Figura 41: Superfície interpolada das temperaturas de referência das estações da NOAA pelo método RST para os aeródromos de referência.

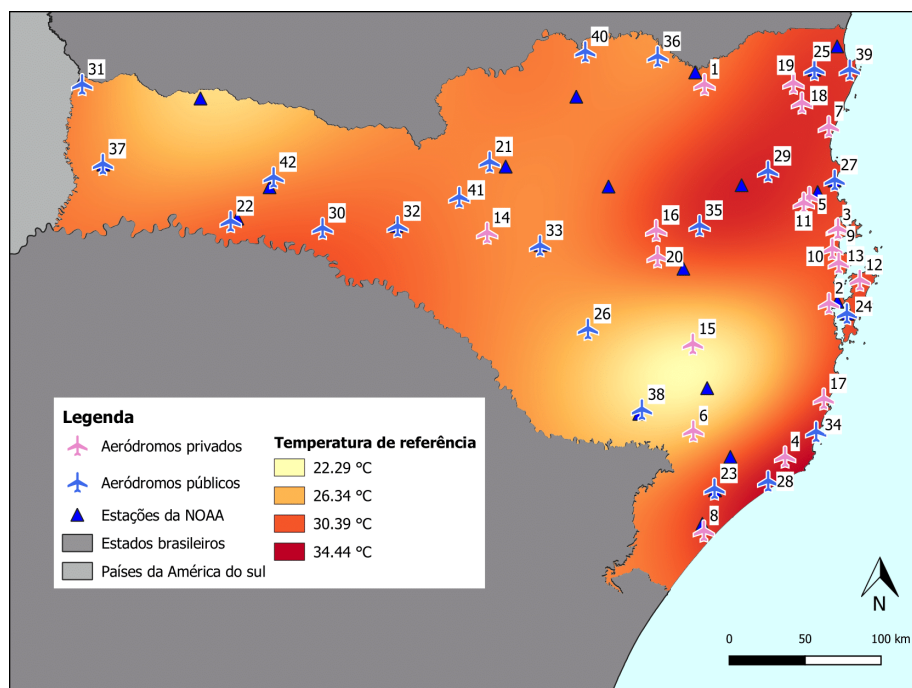


Figura 42: Superfície interpolada das temperaturas de referência das estações da BDMEP pelo método IDW para os aeródromos de referência.

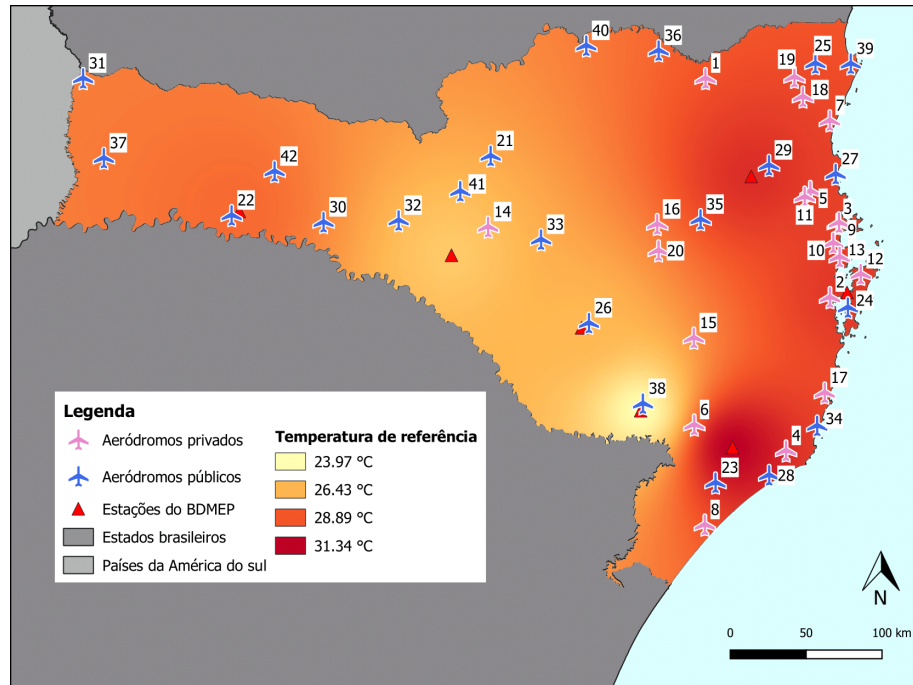
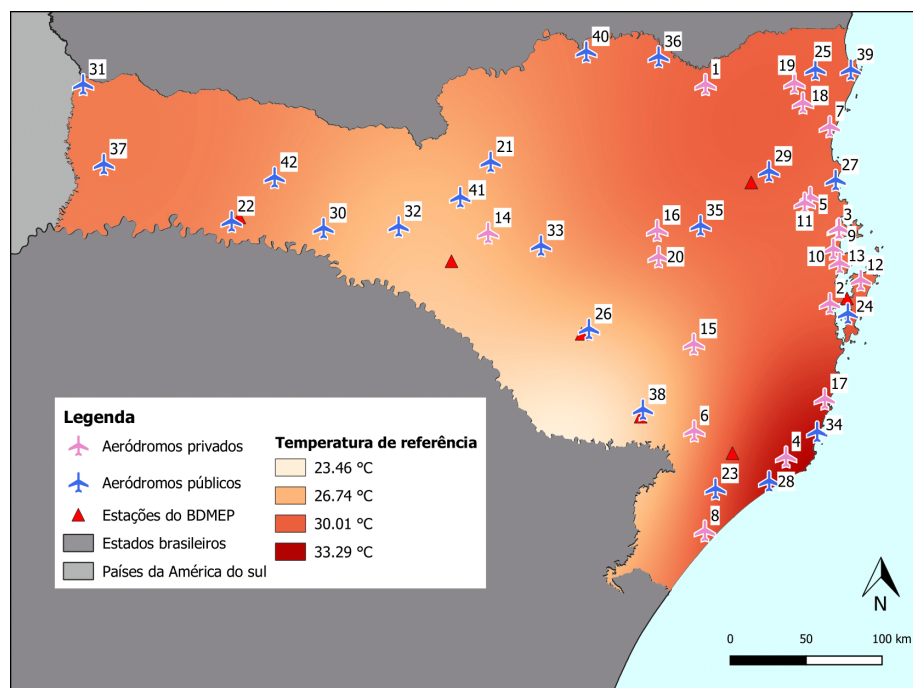


Figura 43: Superfície interpolada das temperaturas de referência das estações da BDMEP pelo método RST para os aeródromos de referência.



A comparação dos dois aeródromos com informações da temperatura de referência são: Aeroporto Hercílio Luz e Ministro Victor Konder, nas cidades de Florianópolis e Navegantes, respectivamente.

Os resultados para as interpolações apresentaram comportamentos semelhantes. Para o Aeroporto Hercílio Luz, os resultados se aproximaram da temperatura de referência dita pela REDEMET. Para o Aeroporto Ministro Victor Konder, o resultado da interpolação se aproximou do divulgado pelo DECEA. As diferenças entre as temperaturas encontradas e as fornecidas pelos órgãos foram menores que 1 grau Celsius.

Tabela 15: Informações e cálculos para o Aeroporto Internacional Hercílio Luz.

Aeroporto Internacional Hercílio Luz	
Código ICAO	SBFL
Nome	Hercílio Luz
Município	Florianópolis
UF	SC
Temperatura de referência DECEA	29 °C
Temperatura de referência dados NOAA por IDW	31,83 °C
Temperatura de referência dados NOAA por RST	30,70 °C
Temperatura de referência dados BDMEP por IDW	29,77 °C
Temperatura de referência dados BDMEP por RST	30,18 °C

Tabela 16: Informações e cálculos para o Aeroporto Ministro Victor Konder.

Aeroporto Ministro Victor Konder	
Código ICAO	SBNF
Nome	Ministro Victor Konder
Município	Navegantes
UF	SC
Temperatura de referência DECEA	29 °C
Temperatura de referência dados NOAA por IDW	31,13 °C
Temperatura de referência dados NOAA por RST	31,57 °C
Temperatura de referência dados BDMEP por IDW	29,49 °C
Temperatura de referência dados BDMEP por RST	29,73 °C

5 Conclusões

O *software* para a automação do procedimento de obtenção da orientação de pistas de pouso e decolagem e da temperatura de referência de sítios aeroportuários apresentou dados que divergem da situação encontrada na realidade, mostrando que outros fatores podem influenciar na escolha da melhor orientação de pista de pouso e decolagem.

No estudo da literatura foi evidenciada a importância que deve ser atribuída à obtenção de bons dados climáticos. O desempenho operacional do aeródromo está ligado diretamente a esta tarefa.

Por meio do desenvolvimento do trabalho, na elaboração do *software*, foi possível identificar diversas formas de trabalhar com dados, otimizando atividades que se resolveriam de maneira mais demorada.

Com o uso das fontes de dados utilizadas neste trabalho, foi possível conhecer as diversas características que os dados proveniente de estações meteorológicas podem ter. Também, quais variáveis são encontradas nos relatórios das estações e quais são disponibilizados pelos responsáveis.

Para os resultados, o desafio de tornar visível os diversos dados, como foram processados e o resultado do processamento. A apresentação deles por meio de tabelas, gráficos e *layouts* de SIG, foi fundamental para facilitar a visualização das informações obtidas.

Outro ponto de observação foram as buscas por dados para referência, onde a indisponibilidade de alguns dados do projeto chamaram a atenção. Até mesmo para os aeródromos de propriedade pública, foram encontrados os dados de temperatura de referência apenas para dois aeródromos dentro da área de estudo. Sendo possível apenas a comparação com dois casos.

Algumas hipóteses puderam ser levantadas acerca dos resultados obtidos. É possível supor que os métodos de interpolação utilizados no trabalho tiveram dados insuficientes ou não são recomendados para interpolação de dados de vento. Também, a orientação de pista pode ser influenciada pela escolha do sítio, e conseqüentemente, da topografia do local.

Por fim, o resultado do *software* desenvolvido neste trabalho pode ser considerado como ferramenta de consulta acadêmica, para o cálculo da orientação de pista de pouso e decolagem de maneira a comparar os resultados com outras ferramentas.

Recomendações para trabalhos futuros

Como sugestão para os futuros trabalhos relacionados ao tema, que não foram desenvolvidos neste trabalho, mas auxiliariam no projeto do aeródromo:

- Aprofundar no estudo de métodos para geração de superfícies interpoladas de dados de vento e temperatura.
- Testar com outros valores de vento de través;
- Desenvolver o cálculo do fator de utilização considerando as áreas de contribuição dentro dos limites de vento de través;
- Testar com dados do Brasil inteiro, e separado para cada estado;
- Aplicar em áreas de estudo de outros países.

Referências

- Abdalla, E.-S. M.; Osman, M. E.; Mohamed, R. N. A. E. monem. Investigation In Selecting The Optimum Airport Runway Orientation With Special Reference To Egyptian Airports. *Journal of Engineering Sciences*, v. 39, n. 6, p. 1261 – 1280, nov. 2011.
- ANAC. *Anuário do Transporte Aéreo 2017*. 2018. Agência Nacional de Aviação Civil. Disponível em: <<https://www.anac.gov.br/assuntos/dados-e-estatisticas/mercado-do-transporte-aereo/ultimas-publicacoes/anuario-do-transporte-aereo-2013-2017>>.
- ANAC. *Aeródromos Cadastrados*. 2019. Disponível em: <<https://www.anac.gov.br/assuntos/setor-regulado/aerodromos/cadastro-de-aerodromos/aerodromos-cadastrados>>.
- ANAC. *Regulamento Brasileiro da Aviação Civil - RBAC 154*. Projeto de Aeródromos, 2019.
- Ashford, N. J.; Mumayiz, S.; Wright, P. H. *Airport Engineering: Planning, Design and Development of 21st Century Airports*. 4. ed. [S.l.: s.n.], 2011. ISBN 9781118005477.
- Bagio, E. et al. Interpolação das variáveis climáticas temperatura do ar e precipitação: Revisão dos métodos mais eficientes. *Geografia*, v. 40, p. 269–289, 08 2015.
- Barbosa, J. P. M. Utilização de método de interpolação para análise e especialização de dados climáticos: O ISG como ferramenta. *CAMINHOS DE GEOGRAFIA*, p. 85–96, fev. 2006. ISSN 1678-6343.
- BDMEP. *Banco de Dados Meteorológicos para Ensino e Pesquisa - BDMEP*. 2019. Disponível em: <<http://www.inmet.gov.br/portal/index.php?r=bdmep/bdmep>>.
- EPAGRI. *Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina - Epagri*. 2019. Disponível em: <<https://www.epagri.sc.gov.br/>>.
- EPAGRI - CIRAM. *Centro de Informações de Recursos Ambientais e de Hidrometeorologia de Santa Catarina - Ciram*. 2019. Disponível em: <<http://ciram.epagri.sc.gov.br>>.
- FAA. *Runway Length Requirements for Airport design - Advisory Circular 150/5325-4B*. [S.l.], 2005.
- FAA. *Airport Design - Advisory Circular 150/5300-13A*. [S.l.], 2014.
- FAA. *Aviation Maintenance Technician Handbook - Airframe, Volume 1*. [S.l.], 2018.
- Horonjeff, R. et al. *Planning & Design of Airports*. Fifth. [S.l.]: McGraw-Hill, 2010.
- ICAO. *Aerodrome Design Manual - Part 1: Runways*. 3. ed. [S.l.], 2006.
- ICAO. *Annex 14: Volume I - Aerodrome Design and Operations*. 8. ed. [S.l.], 2018.
- Kazda, A.; Caves, R. E. *Airport Design and Operation*. 3. ed. [S.l.]: Emerald Group Publishing Limited, 2015.
- McKinsey & Company. *Estudo do Setor de Transporte Aéreo do Brasil: Relatório Consolidado*. Rio de Janeiro, 2010.
- Mello, C. R. et al. Krigagem e inverso do quadrado da distância para interpolação dos parâmetros da equação de chuvas intensas. *Revista Brasileira de Ciência do Solo*, sciELO, v. 27, p. 925 – 933, 10 2003. ISSN 0100-0683.
- Mitas, L.; Mitasova, H. Spatial interpolation. In: _____. *Geographical Information Systems: Principles, Techniques, Management and Applications*. [S.l.: s.n.], 1999. cap. 34, p. 481–492.

NOAA. *Integrated Surface Database - ISD*. 2019. Disponível em: <<https://www.ncdc.noaa.gov/isd>>.

Pamplona, D.; Alves, C. J. P. A correta orientação de uma pista de pouso para o planejamento de um aeroporto. *Revista Científica Linkania Master*, v. 1, n. 6, p. 1,14, 2013. ISSN 2236-6660.

Pandas. *Python Data Analysis Library*. 2019. Disponível em: <<https://pandas.pydata.org/>>.

QGIS. *QGIS Software*. 2019. Disponível em: <<https://www.qgis.org/en/site/>>.

Requests. *Requests*. 2019. Disponível em: <https://requests.readthedocs.io/pt_BR/latest/index.html>.

SQLite. *SQLite*. 2019. Disponível em: <<https://www.sqlite.org/index.html>>.

Vianna, L. F. d. N. et al. Bancos de Dados Meteorológicos: Análise dos Metadados das Estações Meteorológicas no Estado de Santa Catarina, Brasil. *Revista Brasileira de Meteorologia*, scielo, v. 32, p. 53 – 64, 03 2017. ISSN 0102-7786.

Young, S. B.; Wells, A. T. *Airport Planning and Management*. 6. ed. [S.l.]: McGraw-Hill, 2011.

A Arquivo de definição dos pontos centrais das células

```
angle,faixa,x,y
10,1,0.8093,4.5899
10,2,1.4163,8.0323
10,3,2.2945,13.0132
10,4,3.2279,18.3065
10,5,4.1839,23.7283
10,6,5.2201,29.6051
10,7,6.3495,36.0099
20,1,1.5941,4.3797
20,2,2.7896,7.6644
20,3,4.5194,12.4171
20,4,6.3578,17.4679
20,5,8.2408,22.6408
20,6,10.2817,28.2489
20,7,12.5061,34.3603
30,1,2.3304,4.0363
30,2,4.0782,7.0636
[...]
340,4,-6.3578,17.4679
340,5,-8.2408,22.6408
340,6,-10.2817,28.2489
340,7,-12.5061,34.3603
350,1,-0.8093,4.5899
350,2,-1.4163,8.0323
350,3,-2.2945,13.0132
350,4,-3.2279,18.3065
350,5,-4.1839,23.7283
350,6,-5.2201,29.6051
350,7,-6.3495,36.0099
360,1,0.0000,4.6607
360,2,0.0000,8.1563
360,3,0.0000,13.2140
360,4,0.0000,18.5890
360,5,0.0000,24.0944
360,6,0.0000,30.0618
360,7,0.0000,36.5654
```

B Script 00: Selecciona estações do NOAA

Nome: "00_get_station_noaa.py".

```
1 import requests
2 import sqlite3 as lite
3 import os
4 import glob
5 import codecs
6
7
8 diretorio = os.getcwd()
9 URL = 'https://ww1.ncdc.noaa.gov/pub/data/noaa/isd-history.txt'
10
11 page = requests.get(URL)
12
13 isd = str(page.content).replace("\\n", ";").split(";")
14
15 arquivo_csv = "%s\\estacoes_sc.csv" % diretorio
16
17 estacoes = []
18
19 for file in glob.glob(arquivo_csv):
20     c = 0
21     for line in codecs.open(file, "r", "utf-8").readlines():
22         line = line.split(",")
23         if c > 0:
24             usaf = str(line[0].strip().rstrip())
25             statio_name = str(line[1].strip().rstrip())
26             lat = float(line[2].strip().rstrip())
27             lon = float(line[3].strip().rstrip())
28             estacoes.append([ usaf, statio_name, lat, lon ])
29         c += 1
30
31 vec = []
32 for estacao in estacoes:
33     line = []
34     for i in range(len(isd)):
35         l = isd[i].strip()
36         if l[56:64].strip().rstrip() != "" and l[0:6] == estacao[0]:
37             l = isd[i].strip()
38             line.append(l[0:6].strip().rstrip()) # USAF = Air Force station ID. May
39                 contain a letter in the first position
40             line.append(int(l[7:12].strip().rstrip())) # WBAN = NCDC WBAN number
41             line.append(l[13:42].strip().rstrip().replace("&", "").replace("*",
42                 "")) # STATION
43                 NAME
44             line.append(l[43:45].strip().rstrip()) # CTRY = FIPS country ID
45             line.append(l[46:50].strip().rstrip()) # ST = State for US stations
46             line.append(l[51:55].strip().rstrip()) # ICAO = ICAO ID
47             line.append(float(l[56:64].strip().rstrip())) # LAT = Latitude in
48                 thousandths of decimal degrees
49             line.append(float(l[65:73].strip().rstrip())) # LON = Longitude in
50                 thousandths of decimal degrees
51         vec.append(line)
```

```
48 con = lite.connect( "ROP.sqlite" )
49 with con:
50     cur = con.cursor()
51     cur.execute( "DROP TABLE IF EXISTS estacoes" )
52     cur.execute( "CREATE TABLE estacoes (usaf text, wban integer, station_name text,
                    country text, st text, icao text, lat float, lon float, declination float,
                    angulo float, cob float, temp_ref float)" )
53
54     for i in vec:
55         cur.execute("INSERT INTO estacoes (usaf, wban, station_name, country, st,
                    icao, lat, lon) VALUES ('%s', %s, '%s', '%s', '%s', '%s', %s, %s)" % (
56             i[0], i[1], i[2], i[3], i[4], i[5], i[6], i[7]))
57
58 con.close
```

C Script 01: Baixa dados das estações meteorológicas do NOAA

Nome: "01_get_station_data_noaa.py".

```
1 import requests, os, io, glob, gzip, shutil
2 from bs4 import BeautifulSoup
3 import sqlite3 as lite
4
5 diretorio = os.getcwd()
6
7 anos = ['2015', '2016', '2017', '2018', '2019']
8
9 URL = 'https://ww1.ncdc.noaa.gov/pub/data/noaa/'
10
11 headers = {"User-Agent": 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (
    KHTML, like Gecko) Chrome/75.0.3770.100 Safari/537.36'}
12
13 con = lite.connect( "ROP.sqlite" )
14 with con:
15     cur = con.cursor()
16
17     for ano in anos:
18         urlano = "%s/%s" % ( URL, ano )
19         page = requests.get(urlano) #, headers=headers
20         soup = BeautifulSoup(page.content, 'lxml')
21
22         cur.execute( "SELECT usaf FROM estacoes" )
23         estacoes = cur.fetchall()
24
25         for estacao in estacoes:
26             a = soup.findAll('a')
27             for link in a:
28                 s = str(link)
29                 if s.find(str(estacao[0])) > 0:
30                     download = "%s/%s" % ( urlano, link.get_text () )
31                     print(download)
32                     filename = "dados_estacoes\\%s" % (download.split(
33                         "/" )[-1])
34                     with open(filename, "wb") as f:
35                         r = requests.get(download)
36                         f.write(r.content)
37
38         for arquivo in glob.glob(diretorio + "\\dados_estacoes\\*.gz"):
39             with gzip.open(arquivo, 'rb') as f_input:
40                 output_file = arquivo.replace(".gz", ".ish")
41                 print(output_file)
42                 with open(output_file, 'wb') as f_output:
43                     shutil.copyfileobj(f_input, f_output)
44                     f_output.close()
45                 f_input.close()
46 con.close
```

D Script 02: Realiza todos os cálculos

Nome: "02_rop.py".

```
1  # -*- coding: utf-8 -*-
2
3  from math import cos, sin, tan, sqrt, radians, pow
4  import glob, codecs, csv, os, requests
5  import sqlite3 as lite
6  import pandas as pd
7  from pylab import plt
8  import numpy as np
9  from bs4 import BeautifulSoup
10
11  #====CONSTANTES====
12  limites_intervalos_velocidade = [0, 3, 6, 10, 16, 21, 27, 33, 40, 99999999]
13  diretorio = os.getcwd()
14  traves = 10
15
16  estacoes_bdmap = [83897, 83872, 83923, 83920, 83891, 83887, 83883]
17
18  headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (
      KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36'}
19  login_data = {
20      'mUsuario': 'EMAILDELOGINBDMEP',
21      'mCod': 'EMAILDELOGINBDMEP',
22      'mSenha': 'SENHADELOGINBDMEP',
23      'mGerModulo': 'PES',
24      'btnProcesso': ' Acessar '
25  }
26
27  #=====
28
29  def centroides (cur):
30      cur.execute( "DROP TABLE IF EXISTS centroides" )
31      cur.execute( "CREATE TABLE centroides (angle integer, faixa integer, x float, y
      float, qtd integer, percent float)" )
32
33      cur.execute( "DROP TABLE IF EXISTS faixas" )
34      cur.execute( "CREATE TABLE faixas (faixa integer, inicio integer, fim integer)" )
35
36      for j in range(len(limites_intervalos_velocidade) - 1):
37          cur.execute("INSERT INTO faixas (faixa, inicio, fim) VALUES (%s, %s, %s)"
      % (
38              int(j), int(limites_intervalos_velocidade[j]), int(
      limites_intervalos_velocidade[(j+1)]))
39
40      arquivo_csv = diretorio + "\\pontos\\c.csv"
41      with open(arquivo_csv, 'r') as csvfile:
42          plots = csv.reader(csvfile, delimiter=',')
43          c = 0
44          for row in plots:
45              if c > 0:
46                  angle = float(row[0])
```

```

47         faixa = int(row[1])
48         x = float(row[3])
49         y = float(row[2])
50         qtd = float(row[4])
51         percent = float(row[5])
52
53         cur.execute(
54             "INSERT INTO centroides (angle, faixa, x, y, qtd,
55             percent) VALUES (%.2f, %s, %.5f, %.5f, %s, %.5
56             f)" % (
57                 float(angle), int(faixa), float(x), float(y), int(
58                 qtd), float(percent))
59
60         c += 1
61
62         cur.execute("INSERT INTO centroides (angle, faixa, x, y, qtd, percent) VALUES (%.2
63             f, %s, %.5f, %.5f, %s, %.5f)" % (
64             float(0), int(0), float(0), float(0), int(0), float(0)))
65
66         return True
67
68 def ms_to_knots( ms ):
69     return ms * 1.94384
70
71 def conv_ang ( angulo ):
72     if angulo >= 180.:
73         angulo = angulo - 180.
74     if angulo == 180.:
75         angulo = 0.
76     return float(angulo)
77
78 def x(a,c):
79     return ( (-c) / a )
80
81 def ret_coefs( angulo, traves ):
82     angulo = conv_ang(angulo)
83
84     if angulo == 0:
85         a = 0
86         b = 1
87         c = -traves
88     elif angulo == 90:
89         a = 1
90         b = 0
91         c = -traves
92     elif angulo < 90:
93         a = -tan(radians(angulo))
94         b = 1
95         c = -traves/cos(radians(angulo))
96     elif angulo > 90:
97         a = -tan(radians(angulo))
98         b = 1
99         c = -traves/cos(radians(angulo))
100     return a,b,c

```

```

97 def entre_retas ( angulo , traves , x , y ) :
98     retorno = 0
99     a,b,c = ret_coefs(angulo , traves)
100    if angulo == 0:
101        d1 = abs(y - c)
102        d2 = abs(y + c)
103    elif angulo < 90:
104        d1 = (abs(a*x+b*y+c)/(sqrt(pow(a,2)+pow(b,2))))
105        d2 = (abs(a*x+b*y-c)/(sqrt(pow(a,2)+pow(b,2))))
106    elif angulo == 90:
107        d1 = abs(x - c)
108        d2 = abs(x + c)
109    elif angulo > 90:
110        d1 = (abs(a*x+b*y+c)/(sqrt(pow(a,2)+pow(b,2))))
111        d2 = (abs(a*x+b*y-c)/(sqrt(pow(a,2)+pow(b,2))))
112
113    if (float("{0:.5f}".format(round((d1+d2),5))) > (2 * traves)): retorno = 0
114    else: retorno = 1
115    return retorno
116
117 def analisa_vento (estacao , cur):
118
119     dir_arquivos_entrada = diretorio + "\\dados_estacoes"
120     lista_de_arquivos = []
121     for file in glob.glob("%s\\%s-*.ish" % ( dir_arquivos_entrada , estacao )):
122         lista_de_arquivos.append(file)
123
124     cur.execute("DROP TABLE IF EXISTS dados")
125     cur.execute("CREATE TABLE dados (angulo float , velocidade float , tipo character)")
126
127     for arquivo in lista_de_arquivos:
128         for line in codecs.open(arquivo , "r" , "utf-8").readlines():
129             angulo = line[60:63]
130             velocidade = ms_to_knots(float(line[65:69]) / 10.0)
131             tipo = line[64:65]
132             cur.execute(
133                 "INSERT INTO dados ( angulo , velocidade , tipo) VALUES ( %s
134                     , %s , '%s' )" % ( angulo , velocidade , tipo))
135
136     for i in range(35):
137         angulo_ini = (i + 1) * 10 - 5
138         angulo_fim = (i + 1) * 10 + 5
139
140         #print("
141         for j in range(len(limites_intervalos_velocidade) - 1):
142             velocidade_ini = limites_intervalos_velocidade[j]
143             velocidade_fim = limites_intervalos_velocidade[j + 1]
144
145             cur.execute(
146                 "SELECT COUNT(*) FROM dados WHERE angulo>%s AND angulo<=%s
147                     AND velocidade>%s AND velocidade<=%s AND tipo='N' " %
148                     (
149                         angulo_ini , angulo_fim , velocidade_ini ,
150                         velocidade_fim))

```

```

147         quantidade = cur.fetchall()[0][0]
148         if j == 0:
149             cur.execute(
150                 "SELECT qtd FROM centroides WHERE angle=0 AND
151                     faixa=0")
152             qtd = cur.fetchall()[0][0]
153             qtd += quantidade
154             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa =
155                 %s AND angle = %s" % (
156                     int(qtd), int(0), int(0) ))
157         elif j == 8:
158             cur.execute(
159                 "SELECT qtd FROM centroides WHERE angle=%s AND
160                     faixa=%s" % ( int((i+1)*10), int(7) ))
161             qtd = cur.fetchall()[0][0]
162             qtd += quantidade
163             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa =
164                 %s AND angle = %s" % (
165                     int(quantidade), int(7), int((i+1)*10)))
166         else:
167             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa =
168                 %s AND angle = %s" % (
169                     int(quantidade), int(j), int((i+1)*10) ))
170             cur.execute(
171                 "SELECT qtd FROM centroides WHERE angle=%s AND
172                     faixa=%s" % (int((i+1)*10), int(j)))
173             qtd = cur.fetchall()[0][0]
174
175     for j in range(len(limites_intervalos_velocidade) - 1):
176         velocidade_ini = limites_intervalos_velocidade[j]
177         velocidade_fim = limites_intervalos_velocidade[j + 1]
178
179         cur.execute(
180             "SELECT COUNT(*) FROM dados WHERE ((angulo>355 AND angulo<=360) OR
181                 (angulo>0 AND angulo<=5)) AND velocidade>%s AND velocidade<=%
182                 s AND tipo='N'" % (
183                     velocidade_ini, velocidade_fim))
184         quantidade = cur.fetchall()[0][0]
185         if j == 0:
186             cur.execute(
187                 "SELECT qtd FROM centroides WHERE angle=0 AND faixa=0")
188             qtd = cur.fetchall()[0][0]
189             qtd += quantidade
190             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa = %s AND
191                 angle = %s" % (
192                     int(qtd), int(0), int(0)))
193         elif j == 8:
194             cur.execute(
195                 "SELECT qtd FROM centroides WHERE angle=%s AND faixa=%s" %
196                     (int((i + 1) * 10), int(7)))
197             qtd = cur.fetchall()[0][0]
198             qtd += quantidade
199             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa = %s AND
200                 angle = %s" % (

```

```

190             int(qtd), int(7), int(360)))
191         else:
192             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa = %s AND
                           angle = %s" % (
193                 int(quantidade), int(j), float(360)))
194
195         cur.execute("SELECT SUM(qtd) FROM centroides")
196         total = cur.fetchall()[0][0]
197         cur.execute("SELECT angle, faixa, qtd FROM centroides")
198         tabela = cur.fetchall()
199
200         cur.execute("SELECT COUNT(*) FROM centroides")
201         for i in range(cur.fetchall()[0][0]):
202             percent = float((tabela[i][2]/total)*100)
203             cur.execute("UPDATE centroides SET percent = %.5f WHERE faixa = %s AND
                           angle = %.5f" % (
204                 (percent), int(tabela[i][1]), int(tabela[i][0])) )
205         return 1
206
207 def retorna_cobrimento(cur, angulo, traves):
208     cobrimento = 0
209     cur.execute("SELECT angle, faixa, x, y, qtd, percent FROM centroides")
210     tabela = cur.fetchall()
211
212     for i in range(len(tabela)):
213         if entre_retas(angulo, traves, tabela[i][2], tabela[i][3]) == 1:
214             cobrimento += tabela[i][5]
215     return cobrimento
216
217 def melhor_angulo(cur, traves):
218     angulo = 0
219     max_cob = 0
220     for i in range(180):
221         c = retorna_cobrimento(cur, i, float(traves))
222         if c >= max_cob:
223             max_cob = c
224             angulo = i
225     return angulo, max_cob
226
227 def plota_bars(estacao, cur, hab):
228     if (hab):
229         cur.execute("SELECT angle, faixa, x, y, qtd, percent FROM centroides")
230         tabela = cur.fetchall()
231         x = []
232         y = []
233         for row in range(len(tabela)):
234             x.append(float(tabela[row][2]))
235             y.append(float(tabela[row][3]))
236
237         cobs = []
238         angs = []
239         for i in range(180):
240             c = retorna_cobrimento(cur, (i+1), traves)
241             angs.append(i+1)

```

```

242         cobs.append(float(c))
243         plt.title(estacao)
244         plt.bar(angs, cobs, align='center', color= '#00BFFF')
245         plt.xticks(angs,angs,rotation=90, size='small')
246         plt.show()
247     return True
248
249 def get_declination(lat, lon):
250     headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit
251               /537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36'}
252     with requests.Session() as s:
253         norte_sul = ("S" if lat < 0 else "N")
254         leste_oeste = ("W" if lon < 0 else "E")
255         lat = lat * -1.
256         lon = lon * -1.
257         login_page = "https://www.ngdc.noaa.gov/geomag-web/calculators/
258                       calculateDeclination?browserRequest=true&magneticComponent=d&lat1=%s&
259                       lat1Hemisphere=%s&lon1=%s&lon1Hemisphere=%s&model=WMM&startYear=2019&
260                       startMonth=10&startDay=3&resultFormat=html&ajax=true" % (
261                           lat, norte_sul, lon, leste_oeste
262                       )
263
264         login = s.get(login_page, headers=headers)
265         soup = BeautifulSoup(login.content, 'lxml')
266         declination = str(soup.findAll('td')[10]).split()[2].replace("°", "")
267
268     return -float(declination)
269
270 def csv_output_noaa (cur, plotar):
271     retorno = False
272     try:
273         print("Processando para as estacoes do NOAA")
274
275         output = codecs.open( "SC_NOAA.csv", "w", "utf-8" )
276         output.write( "USAF,STATION_NAME,ICAO,LAT,LON,DECLINATION,ANGULO,
277                       COBRIMENTO,TEMPERATURA_REFERENCIA\n" )
278
279         cur.execute("SELECT * FROM estacoes WHERE temp_ref < 999.9 AND USAF !=
280                       838870")
281         tabela = cur.fetchall()
282         for estacao in tabela:
283             print(estacao[0])
284
285             centroides(cur)
286             analisa_vento(estacao[0], cur)
287
288             #PLOTA GRAFICO DE BARRAS?
289             # plotar = False
290             plota_bars("%s-%s" % (estacao[0], estacao[2]), cur, plotar)
291
292             declination = get_declination( estacao[6], estacao[7] )
293
294             #PLOTA ANGULO COM MELHOR COBRIMENTO

```

```

290         ang_cob = melhor_angulo(cur, traves)
291
292         if ang_cob[0] != 179 and ang_cob[1] != 100:
293             cur.execute("UPDATE estacoes SET angulo = %s , cob = %s,
                declination = %s WHERE usaf = '%s' " % (ang_cob[0],
                ang_cob[1], declination, estacao[0]))
294
295             retorno = True
296     except Exception as erro:
297         print ("ERRO NO PROCESSO DO NOAA: %s " % erro)
298         retorno = False
299     finally:
300         output.close
301         return retorno
302
303 def calcula_temp_ref_noaa(cur):
304
305     print("Calculando Temp_ref para as estacoes do NOAA")
306
307     dir_arquivos_entrada = "%s\\dados_estacoes" % diretorio
308
309     cur.execute( "DROP TABLE IF EXISTS dados_temp" )
310     cur.execute( "CREATE TABLE dados_temp (estacao string, dia integer, mes string,
311         ano string, hora integer, temperatura float)" )
312     cur.execute( "CREATE UNIQUE INDEX dados_temp_idx ON dados_temp ( estacao, hora,
313         dia, mes, ano, temperatura )" ) # Junto com o posterior INSERT OR IGNORE, para evitar
314         INSERTS de dados duplicados, caso houverem.
315
316     cur.execute( "DROP TABLE IF EXISTS max_temps_dias" )
317     cur.execute( "CREATE TABLE max_temps_dias (estacao string, dia integer, mes string
318         , ano string, max_temp float)" )
319
320     cur.execute( "DROP TABLE IF EXISTS temp_medias_mes" )
321     cur.execute( "CREATE TABLE temp_medias_mes (estacao string, mes string, ano string
322         , media_mes float)" )
323
324     cur.execute("SELECT distinct usaf FROM estacoes")
325     tabela = cur.fetchall()
326
327     for usaf in tabela:
328         estacao = usaf[0]
329
330         lista_de_arquivos = []
331         for file in glob.glob("%s\\%s-*.ish" % ( dir_arquivos_entrada, estacao ) )
332             :
333             lista_de_arquivos.append(file)
334
335         for arquivo in lista_de_arquivos:
336
337             for line in codecs.open(arquivo, "r", "utf-8").readlines():
338
339                 dia = int( line[21:23] )
340                 mes = int( line[19:21] )
341                 ano = int( line[15:19] )
342                 hora = int( line[23:27] )

```

```

337
338         temp = float( line[87:92] ) / 10.0
339
340         if temp != 999.9:
341             cur.execute("INSERT OR IGNORE INTO dados_temp (
342                 estacao, dia, mes, ano, hora, temperatura)
343                 VALUES ('%s', %s, '%s', '%s', %s, %s)" % (
344                     estacao, dia, mes, ano, hora, temp ) )
345
346             # inseri na tabela as maximas temperaturas diarias
347             cur.execute( """INSERT INTO max_temps_dias ( estacao, dia, mes, ano,
348                 max_temp
349
350                 SELECT estacao, dia, mes, ano, MAX(
351                     temperatura) as max_temp
352                 FROM dados_temp WHERE estacao = '%s' GROUP
353                 BY estacao, ano, mes, dia""" %
354                     estacao )
355
356             # inseri na tabela as medias mensais
357             cur.execute( """INSERT INTO temp_medias_mes (estacao, mes, ano, media_mes)
358                 SELECT estacao, mes, ano, AVG(temperatura)
359                 AS media_mes
360                 FROM dados_temp WHERE estacao = '%s'
361                 GROUP BY estacao, ano, mes""" % estacao)
362
363         try:
364             cur.execute( "SELECT mes, ano, media_mes FROM temp_medias_mes
365                 WHERE estacao = '%s' ORDER BY media_mes DESC" % estacao)
366             resultado = cur.fetchall()
367             # print(resultado)
368             maxMes = resultado[0][0]
369             maxAno = resultado[0][1]
370
371             cur.execute("SELECT AVG(max_temp) as media_max_temp FROM
372                 max_temps_dias WHERE estacao = '%s' AND mes = '%s' AND ano =
373                 '%s' " % (estacao, str(maxMes), str(maxAno)))
374             resultado = cur.fetchall()
375             mediaMaxTemp = resultado[0][0]
376
377             cur.execute("UPDATE estacoes SET temp_ref = %s WHERE usaf = '%s' "
378                 % ( mediaMaxTemp, estacao ))
379         except Exception :
380             cur.execute("UPDATE estacoes SET temp_ref = 999.9 WHERE usaf = '%s'
381                 ' " % ( estacao ))
382
383     return True
384
385 #=====
386 #=====
387 # BDMAP
388 def analisa_vento_bdmap (estacao, cur):
389
390     cur.execute("DROP TABLE IF EXISTS dados")

```



```

378 cur.execute("CREATE TABLE dados (estacao integer, timestamp date, angulo float,
379 velocidade float)")
379 cur.execute( "CREATE UNIQUE INDEX dados_bdmap_idx ON dados ( estacao, timestamp,
angulo )" ) # Junto com o posterior INSERT OR IGNORE, para evitar INSERTS de dados
duplicados, caso houverem.
380
381 with requests.Session() as s:
382     login_page = "http://www.inmet.gov.br/projetos/rede/pesquisa/inicio.php?&
mCod=EMAILDELOGIN&mGerModulo=PES&mSenha=SENHADELOGIN&btnProcesso=%20
Acessar%20"
383
384     nome_estacao = ""
385     cod_estacao = ""
386     base = []
387     #login
388     login = s.get(login_page, headers=headers) #, headers=headers
389     soup = BeautifulSoup(login.content, 'lxml')
390     login_data['mGerModulo'] = soup.find('input', attrs={'name':'mGerModulo'})
['value']
391     s.post(login_page, data=login_data, headers=headers)
392
393     data = "http://www.inmet.gov.br/projetos/rede/pesquisa/gera_serie_txt.php
?&mRelEstacao=%s&btnProcesso=serie&mRelDtInicio=01/01/2015&mRelDtFim
=31/10/2019&mAtributos=,,,,,,1,1,,,,,, " % (estacao)
394     data_page = s.get(data, headers=headers)
395     soup = BeautifulSoup(data_page.content, 'lxml')
396     txt = str(soup.find('pre')).split('\n')
397
398     cont_linha = 0
399     for line in txt:
400         # print(line)
401         if cont_linha == 3:
402             nome_estacao = line.replace(" ", "").split(":")
403             cod_estacao = nome_estacao[2].replace("","")
404             nome_estacao = nome_estacao[1].split("(")[0]
405         if cont_linha == 4:
406             lat = float(line.replace(" ", "").split(":")[1])
407         if cont_linha == 5:
408             lon = float(line.replace(" ", "").split(":")[1])
409
410         elif cont_linha > 16:
411             try:
412                 inf = line.split(";")
413                 date = inf[1].split("/")
414                 hour = inf[2][0:2]
415                 minute = inf[2][2:4]
416                 timestamp = "%s-%s-%s %s:%s:00" % ( date[2], date
[1], date[0], hour, minute)
417                 vel = ms_to_knots(float(inf[4]))
418                 dirr = float(inf[3]) * 10
419                 if dirr != 0 and dirr != 990:
420                     base.append([cod_estacao, nome_estacao.
lower(), timestamp, dirr, vel])
421             except Exception:

```

```

422                 break
423
424                 cont_linha +=1
425
426                 for l in base:
427                     cur.execute("INSERT INTO dados ( estacao, timestamp, angulo,
428                                 velocidade ) VALUES (%s, '%s', %s, %s)" % ( l[0], l[2], l[3],
429                                     l[4] ))
428
429                 # ANALISE
430                 for i in range(35):
431                     angulo_ini = (i + 1) * 10 - 5
432                     angulo_fim = (i + 1) * 10 + 5
433
434                     for j in range(len(limites_intervalos_velocidade) - 1):
435                         velocidade_ini = limites_intervalos_velocidade[j]
436                         velocidade_fim = limites_intervalos_velocidade[j + 1]
437
438                         cur.execute(
439                             "SELECT COUNT(*) FROM dados WHERE angulo>%s AND angulo<=%s
440                                 AND velocidade>%s AND velocidade<=%s" % (
441                                     angulo_ini, angulo_fim, velocidade_ini,
442                                     velocidade_fim))
441                         quantidade = cur.fetchall()[0][0]
442                         if j == 0:
443                             cur.execute(
444                                 "SELECT qtd FROM centroides WHERE angle=0 AND
445                                     faixa=0")
445                             qtd = cur.fetchall()[0][0]
446                             qtd += quantidade
447                             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa =
448                                     %s AND angle = %s" % (
449                                         int(qtd), int(0), int(0) ))
449                         elif j == 8:
450                             cur.execute(
451                                 "SELECT qtd FROM centroides WHERE angle=%s AND
452                                     faixa=%s" % ( int((i+1)*10), int(7) ))
452                             qtd = cur.fetchall()[0][0]
453                             qtd += quantidade
454                             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa =
455                                     %s AND angle = %s" % (
456                                         int(quantidade), int(7), int((i+1)*10)))
456                         else:
457                             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa =
458                                     %s AND angle = %s" % (
459                                         int(quantidade), int(j), int((i+1)*10) ))
459                             cur.execute(
460                                 "SELECT qtd FROM centroides WHERE angle=%s AND
461                                     faixa=%s" % (int((i+1)*10), int(j)))
461                             qtd = cur.fetchall()[0][0]
462
463                 for j in range(len(limites_intervalos_velocidade) - 1):
464                     velocidade_ini = limites_intervalos_velocidade[j]
465                     velocidade_fim = limites_intervalos_velocidade[j + 1]

```

```

466
467         cur.execute(
468             "SELECT COUNT(*) FROM dados WHERE ((angulo>355 AND angulo<=360) OR
              (angulo>0 AND angulo<=5)) AND velocidade>%s AND velocidade<=%
              s " % (
469                 velocidade_ini, velocidade_fim))
470         quantidade = cur.fetchall()[0][0]
471         if j == 0:
472             cur.execute(
473                 "SELECT qtd FROM centroides WHERE angle=0 AND faixa=0")
474             qtd = cur.fetchall()[0][0]
475             qtd += quantidade
476             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa = %s AND
              angle = %s" % (
477                 int(qtd), int(0), int(0)))
478         elif j == 8:
479             cur.execute(
480                 "SELECT qtd FROM centroides WHERE angle=%s AND faixa=%s" %
              (int((i + 1) * 10), int(7)))
481             qtd = cur.fetchall()[0][0]
482             qtd += quantidade
483             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa = %s AND
              angle = %s" % (
484                 int(qtd), int(7), int(360)))
485         else:
486             cur.execute("UPDATE centroides SET qtd = %s WHERE faixa = %s AND
              angle = %s" % (
487                 int(quantidade), int(j), float(360)))
488
489         cur.execute("SELECT SUM(qtd) FROM centroides")
490         total = cur.fetchall()[0][0]
491         cur.execute("SELECT angle, faixa, qtd FROM centroides")
492         tabela = cur.fetchall()
493
494         cur.execute("SELECT COUNT(*) FROM centroides")
495         for i in range(cur.fetchall()[0][0]):
496             percent = float((tabela[i][2]/total)*100)
497
498             cur.execute("UPDATE centroides SET percent = %.5f WHERE faixa = %s AND
              angle = %.5f" % (
499                 (percent), int(tabela[i][1]), int(tabela[i][0])) )
500
501         return lat, lon, nome_estacao.lower()
502
503 def calcula_temp_ref_bdmap(estacao, cur):
504
505
506         cur.execute("DROP TABLE IF EXISTS dados_temp")
507         cur.execute("CREATE TABLE dados_temp (estacao string, dia integer, mes string,
              ano string, hora integer, temperatura float)")
508         cur.execute("CREATE UNIQUE INDEX dados_temp_idx ON dados_temp ( estacao, hora,
              dia, mes, ano )" ) # Junto com o posterior INSERT OR IGNORE, para evitar INSERTS de dados
              duplicados, caso houverem.
509
510         cur.execute("DROP TABLE IF EXISTS max_temps_dias")

```

```

511 cur.execute( "CREATE TABLE max_temps_dias (estacao string, dia integer, mes string
           , ano string, max_temp float)" )
512
513 cur.execute( "DROP TABLE IF EXISTS temp_medias_mes" )
514 cur.execute( "CREATE TABLE temp_medias_mes (estacao string, mes string, ano string
           , media_mes float)" )
515
516 with requests.Session() as s:
517     login_page = "http://www.inmet.gov.br/projetos/rede/pesquisa/inicio.php?&
           mCod=EMAILDELOGIN&mGerModulo=PES&mSenha=SENHADELOGIN&btnProcesso=%20
           Acessar%20"
518
519     nome_estacao = ""
520     cod_estacao = ""
521     base = []
522     #login
523     login = s.get(login_page, headers=headers) #, headers=headers
524     soup = BeautifulSoup(login.content, 'lxml')
525     login_data['mGerModulo'] = soup.find('input', attrs={'name':'mGerModulo'})
           ['value']
526     s.post(login_page, data=login_data, headers=headers)
527
528     data = "http://www.inmet.gov.br/projetos/rede/pesquisa/gera_serie_txt.php
           ?&mRelEstacao=%s&btnProcesso=serie&mRelDtInicio=01/01/2015&mRelDtFim
           =31/10/2019&mAtributos=1,,,,,,,,,,,,," % (estacao)
529
530     data_page = s.get(data, headers=headers)
531     soup = BeautifulSoup(data_page.content, 'lxml')
532
533     txt = str(soup.find('pre')).split('\n')
534
535     cont_linha = 0
536     for line in txt:
537         # print(line)
538         if cont_linha == 3:
539             nome_estacao = line.replace(" ", "").split(":")
540             cod_estacao = nome_estacao[2].replace(")", "")
541             nome_estacao = nome_estacao[1].split("(")[0]
542
543         elif cont_linha > 16:
544             try:
545                 inf = line.split(";")
546                 date = inf[1].split("/")
547                 dia = date[0]
548                 mes = date[1]
549                 ano = date[2]
550                 hora = inf[2][0:4]
551                 temp = float(inf[3])
552
553                 base.append([cod_estacao, nome_estacao.lower(),
                               dia, mes, ano, hora, temp])
554             except Exception:
555                 break

```

```

556
557             cont_linha +=1
558
559     for l in base:
560         cur.execute("INSERT OR IGNORE INTO dados_temp (estacao, dia, mes, ano,
561                     hora, temperatura) VALUES ('%s', %s, '%s', '%s', %s, %s)" %
562                     ( l[0], l[2], l[3], l[4], l[5], l[6] ) )
563
564     # inseri na tabela as maximas temperaturas diarias
565     cur.execute( """INSERT INTO max_temps_dias ( estacao, dia, mes, ano, max_temp)
566                 SELECT estacao, dia, mes, ano, MAX(temperatura) as
567                     max_temp
568                 FROM dados_temp WHERE estacao = '%s' GROUP BY
569                     estacao, ano, mes, dia""" % estacao )
570
571     # inseri na tabela as medias mensais
572     cur.execute( """INSERT INTO temp_medias_mes (estacao, mes, ano, media_mes)
573                 SELECT estacao, mes, ano, AVG(temperatura) AS
574                     media_mes
575                 FROM dados_temp WHERE estacao = '%s'
576                 GROUP BY estacao, ano, mes""" % estacao)
577
578     try:
579         cur.execute( "SELECT mes, ano, media_mes FROM temp_medias_mes WHERE
580                     estacao = '%s' ORDER BY media_mes DESC" % estacao)
581         resultado = cur.fetchall()
582         maxMes = resultado[0][0]
583         maxAno = resultado[0][1]
584
585         cur.execute("SELECT AVG(max_temp) as media_max_temp FROM max_temps_dias
586                     WHERE estacao = '%s' AND mes = '%s' AND ano = '%s' " % (estacao, str(
587                         maxMes), str(maxAno)))
588         resultado = cur.fetchall()
589         mediaMaxTemp = resultado[0][0]
590     except Exception:
591         mediaMaxTemp = 0
592
593     return mediaMaxTemp
594
595 def csv_output_bdmap (cur, plotar):
596     retorno = False
597     try:
598         output = codecs.open( "SC_BDMAP.csv", "w", "utf-8" )
599         output.write( "ESTACAO,NOME_ESTACAO,LAT,LON,DECLINATION,ANGULO,COBRIMENTO,
600                     TEMPERATURA_REFERENCIA\n" )
601
602         cur.execute("DROP TABLE IF EXISTS verificaEstacoesBDMEP")
603         cur.execute("CREATE TABLE verificaEstacoesBDMEP (estacao text, nome text,
604                     ano integer, qtd integer)")
605
606         print("Processamento BDMAP")
607         for estacao in estacoes_bdmap:
608             print(estacao)

```


E *Script 03: Cria shapefile dos resultados das estações*

Nome: "03_create_shp_stations.py".

```
1  # -*- coding: utf-8 -*-
2
3  import pandas as pd
4  import os, glob
5  from shapely.geometry import Point
6  from geopandas import GeoDataFrame
7
8  diretorio = os.getcwd()
9
10 lista_de_arquivos = []
11 for file in glob.glob("%s\\SC_*.csv" % ( diretorio )):
12     lista_de_arquivos.append(file)
13
14
15 for estacao in lista_de_arquivos:
16     df = pd.read_csv(estacao)
17     nome = estacao.split("\\")[ -1].split(".csv")[0]
18
19     geometry = [Point(xy) for xy in zip(df.LON, df.LAT)]
20
21     #32722 = WGS84 UTM22S
22     #4326 = WGS84
23     crs = {'init': 'epsg:4326'}
24     geo_df = GeoDataFrame(df, crs=crs, geometry=geometry)
25     shp_utm = geo_df.to_crs(epsg=32722)
26
27     save_directory = "%s\\grass_script\\estacoes_utm\\estacoes_UTM_%s.shp" % ( diretorio,
28                                     nome)
29     shp_utm.to_file(driver='ESRI Shapefile', filename=save_directory)
```

F *Script 04: Selecciona os aeródromos da área de estudo e gera um shapefile*

Nome: "04_get_aeros.py".

```
1  # -*- coding: utf-8 -*-
2
3  import pandas as pd
4  import glob
5  import xlrd
6  import sqlite3 as lite
7  import os
8  import codecs
9  import unidecode
10
11 from unicodedata import normalize
12 from shapely.geometry import Point
13 from geopandas import GeoDataFrame
14
15 #=====
16
17 def detox (s):
18     s = s.replace(" ", "_")
19     s = unidecode.unidecode(s)
20     return s
21
22 def removerCaracteresEspeciais (text) :
23     return normalize('NFKD', text).encode('ASCII', 'ignore').decode('ASCII')
24
25 #=====
26
27 directorio = os.getcwd()
28
29 for name in glob.glob('%s\\aerodromos\\*.xls' % directorio):
30
31     name_parts = name.split("\\")
32     file_name = name_parts[-1]
33     print (file_name)
34     xls = xlrd.open_workbook(name, on_demand=True)
35     sheet_list = xls.sheet_names()
36     for sheet in sheet_list:
37
38         if 'Aer' in sheet:
39
40             output = "%s\\aerodromos\\SC_%s.csv" % (directorio, detox(file_name
41                 .split("_")[0]))
42             print (output)
43
44             df = pd.read_excel(name, sheet_name = sheet)
45
46             catarina_santa = df.UF == 'SC'
47
48             sc = df[catarina_santa]
```



```

48         sc.to_csv(output, sep=";", index = False)
49
50     cod_icao = []
51     nome = []
52     municipio = []
53     uf = []
54     lat = []
55     lon = []
56     designacao = []
57     propriedade = []
58
59     for name in glob.glob('%s\\aerodromos\\SC*.csv' % diretorio):
60         publico = False
61         if "Publicos" in name:
62             publico = True
63         conta = 1
64         for line in open(name, "r", encoding='utf8').readlines():
65
66             if (publico and conta > 3):
67                 partes = line.split(";")
68                 cod_icao.append(partes[0])
69                 nome.append(removerCaracteresEspeciais(partes[2].lower()))
70                 municipio.append(removerCaracteresEspeciais(partes[3].lower()))
71                 uf.append(partes[4])
72
73                 lat_processando = (partes[5]).replace("\\'", "").replace("Â°", "").
74                     split(" ")
75                 sinal_lat = -1 if 'S' in lat_processando[3] else 1
76                 lat_processado = sinal_lat*(int(lat_processando[0]) + int(
77                     lat_processando[1])/60 + int(lat_processando[2])/3600)
78                 lat.append(lat_processado)
79
80                 lon_processando = (partes[6]).replace("\\'", "").replace("Â°", "").
81                     split(" ")
82                 sinal_lon = -1 if 'W' in lon_processando[3] else 1
83                 lon_processado = sinal_lon*(int(lon_processando[0]) + int(
84                     lon_processando[1])/60 + int(lon_processando[2])/3600)
85                 lon.append(lon_processado)
86
87                 designacao.append(partes[9].strip())
88                 propriedade.append("publico")
89
90             if (not publico and conta > 1):
91                 partes = line.split(";")
92                 cod_icao.append(partes[0])
93                 nome.append(removerCaracteresEspeciais(partes[2].lower()))
94                 municipio.append(removerCaracteresEspeciais(partes[3].lower()))
95                 uf.append(partes[4])
96
97                 lat_processando = (partes[5]).replace("\\'", "").replace("Â°", "").
98                     replace("\\\"", "").split(" ")
99                 sinal_lat = -1 if 'S' in lat_processando[3] else 1
100                lat_processado = sinal_lat*(int(lat_processando[0]) + int(

```

```

121         lat_processando[1])/60 + int(lat_processando[2])/3600)
97     lat.append(lat_processado)
98
99     lon_processando = (partes[6]).replace("\'", "").replace("Â°", "").
100         replace("\'", "").split(" ")
101     sinal_lon = -1 if 'W' in lon_processando[3] else 1
102     lon_processado = sinal_lon*(int(lon_processando[0]) + int(
103         lon_processando[1])/60 + int(lon_processando[2])/3600)
104     lon.append(lon_processado)
105
106     designacao.append(partes[9].strip())
107     propriedade.append("privado")
108
109     conta +=1
110
111 aerodromos = pd.DataFrame({
112     'cod_icao':cod_icao,
113     'nome':nome,
114     'municipio':municipio,
115     'uf':uf,
116     'lat':lat,
117     'lon':lon,
118     'designacao':designacao,
119     'propriedade':propriedade
120 })
121
122 geometry = [Point(xy) for xy in zip(aerodromos.lon, aerodromos.lat)]
123
124 #32722 = WGS84 UTM22S
125 #4326 = WGS84
126 crs = {'init': 'epsg:4326'}
127 geo_df = GeoDataFrame(aerodromos, crs=crs, geometry=geometry)
128 shp_utm = geo_df.to_crs(epsg=32722)
129
130 save_directory = "%s\\aerodromos\\shapefile\\aerodromos_existentes_UTM.shp" % diretorio
131 shp_utm.to_file(driver='ESRI Shapefile', filename=save_directory)
132
133 output = "%s\\aerodromos\\aerodromos_sc.csv" % (diretorio)
134
135 aerodromos = aerodromos.drop('geometry', axis=1)
136 aerodromos['coordenadas'] = shp_utm.geometry
137
138 aerodromos.to_csv(output, sep=";", index = False, encoding='utf8')

```

G *Script 05*: Interpola os dados e determina a orientação e temperatura para os aeródromos de referência

Nome: "05_load_data_grass.py".

```
1  # -*- coding: utf-8 -*-
2
3  # ===== IMPORTS =====
4  import os
5  import sys
6  import datetime
7  import fnmatch
8  import math
9  import commands
10 import subprocess
11 import codecs
12 from subprocess import PIPE
13
14 # GRASS
15
16 from grass.pygrass.modules import Module
17 from grass.pygrass.modules.shortcuts import general as g
18 from grass.pygrass.gis.region import Region
19 import grass.pygrass.modules.interface.read as Read
20 import grass.script as grass
21
22
23 # Rodar dentro do GRASS GIS
24
25 # =====
26 # ===== DADOS DE ENTRADA =====
27
28 diretorio = os.getcwd()
29
30
31
32 diretorio_grass = "%s\\grass_script" % diretorio
33
34 estacoes_utm_sc_noaa = "estacoes_UTM_SC_NOAA"
35 estacoes_utm_sc_bdmap = "estacoes_UTM_SC_BDMAP"
36 sc_utm = "estado_sc"
37 estacoes_dir_noaa = "%s\\estacoes_utm\\%s.shp" % ( diretorio_grass , estacoes_utm_sc_noaa )
38 estacoes_dir_bdmap = "%s\\estacoes_utm\\%s.shp" % ( diretorio_grass , estacoes_utm_sc_bdmap
39 )
40 estado_dir = "%s\\estado\\%s.shp" % ( diretorio_grass , sc_utm )
41
42
43 # ===== FUNCOES =====
44
45 def cleanGRASSVector():
46     Module('g.remove', flags='f', name=estacoes_utm_sc_noaa, type='vector')
47     Module('g.remove', flags='f', name=estacoes_utm_sc_bdmap, type='vector')
48     Module('g.remove', flags='f', name=sc_utm, type='vector')
49     Module('g.remove', flags='f', name='MASK', type='raster')
```

```

48     Module('g.remove', flags='f', name='sc_mask', type='raster')
49
50     #angulo da pista
51     Module('g.remove', flags='f', name='sc_estacoes_noaa_idw_rw', type='raster')
52     Module('g.remove', flags='f', name='sc_estacoes_bdmap_idw_rw', type='raster')
53     Module('g.remove', flags='f', name='sc_estacoes_noaa_rst_rw', type='raster')
54     Module('g.remove', flags='f', name='sc_estacoes_bdmap_rst_rw', type='raster')
55
56     #Temperatura
57     Module('g.remove', flags='f', name='sc_estacoes_noaa_idw_temp', type='raster')
58     Module('g.remove', flags='f', name='sc_estacoes_bdmap_idw_temp', type='raster')
59     Module('g.remove', flags='f', name='sc_estacoes_noaa_rst_temp', type='raster')
60     Module('g.remove', flags='f', name='sc_estacoes_bdmap_rst_temp', type='raster')
61
62     # =====
63
64     cleanGRASSVector()
65
66     #importando vetores
67     ogrimport = Module('v.in.ogr')
68     ogrimport( estado_dir, output=sc_utm )
69     ogrimport( estacoes_dir_noaa , output=estacoes_utm_sc_noaa)
70     ogrimport( estacoes_dir_bdmap , output=estacoes_utm_sc_bdmap)
71
72     # definindo regioao
73     Module('g.region', nsres=50,ewres=50,vector=sc_utm)
74
75     #mascara
76     Module('v.to.rast', input=sc_utm, type=['point','line','boundary','area'], output='sc_mask
       ', use='attr', attribute_column='cat')
77     Module('r.mask', raster='sc_mask')
78
79     #interpola para o angulo da pista
80     Module('v.surf.idw', input=estacoes_utm_sc_noaa, column='ANGULO', output='
       sc_estacoes_noaa_idw_rw')
81     Module('v.surf.idw', input=estacoes_utm_sc_bdmap, column='ANGULO', output='
       sc_estacoes_bdmap_idw_rw')
82     Module('v.surf.rst', input=estacoes_utm_sc_noaa, mask='MASK', zcolumn='ANGULO', elevation=
       'sc_estacoes_noaa_rst_rw')
83     Module('v.surf.rst', input=estacoes_utm_sc_bdmap, mask='MASK', zcolumn='ANGULO', elevation
       ='sc_estacoes_bdmap_rst_rw')
84
85     #interpola para a temperatura
86     Module('v.surf.idw', input=estacoes_utm_sc_noaa, column='TEMPERATUR', output='
       sc_estacoes_noaa_idw_temp')
87     Module('v.surf.idw', input=estacoes_utm_sc_bdmap, column='TEMPERATUR', output='
       sc_estacoes_bdmap_idw_temp')
88     Module('v.surf.rst', input=estacoes_utm_sc_noaa, mask='MASK', zcolumn='TEMPERATUR',
       elevation='sc_estacoes_noaa_rst_temp')
89     Module('v.surf.rst', input=estacoes_utm_sc_bdmap, mask='MASK', zcolumn='TEMPERATUR',
       elevation='sc_estacoes_bdmap_rst_temp')
90
91     # salvar RASTERS
92     # angulo

```

```

93 output_diretorio = "%s\\imagens\\noaa_idw_rw.tif" % (diretorio,)
94 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_noaa_idw_rw', output=
    output_diretorio, format='GTiff', type='Float32')
95 output_diretorio = "%s\\imagens\\bdmep_idw_rw.tif" % (diretorio,)
96 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_bdmap_idw_rw', output=
    output_diretorio, format='GTiff', type='Float32')
97 output_diretorio = "%s\\imagens\\noaa_rst_rw.tif" % (diretorio,)
98 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_noaa_rst_rw', output=
    output_diretorio, format='GTiff', type='Float32')
99 output_diretorio = "%s\\imagens\\bdmep_rst_rw.tif" % (diretorio,)
100 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_bdmap_rst_rw', output=
    output_diretorio, format='GTiff', type='Float32')
101
102 # temp
103 output_diretorio = "%s\\imagens\\noaa_idw_temp.tif" % (diretorio,)
104 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_noaa_idw_temp', output=
    output_diretorio, format='GTiff', type='Float32')
105 output_diretorio = "%s\\imagens\\bdmep_idw_temp.tif" % (diretorio,)
106 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_bdmap_idw_temp', output
    =output_diretorio, format='GTiff', type='Float32')
107 output_diretorio = "%s\\imagens\\noaa_rst_temp.tif" % (diretorio,)
108 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_noaa_rst_temp', output=
    output_diretorio, format='GTiff', type='Float32')
109 output_diretorio = "%s\\imagens\\bdmep_rst_temp.tif" % (diretorio,)
110 Module('r.out.gdal', flags='f', overwrite=True, input='sc_estacoes_bdmap_rst_temp', output
    =output_diretorio, format='GTiff', type='Float32')
111
112 # Lendo
113 aerodromos = []
114 for line in open("%s\\aerodromos\\aerodromos_sc.csv" % diretorio, "r").readlines():
115     partes = line.replace("POINT (", "").replace("\n", "").split(";")
116     aerodromos.append(partes)
117
118 # Escrevendo
119 output_name = "%s\\aerodromos\\aerodromos_sc.csv" % diretorio
120 output = codecs.open(output_name, "w", "utf-8")
121
122 conta_linha = 0
123 for line in aerodromos:
124     coords = line[-1].split()
125
126     if conta_linha == 0:
127         head = "%s;angulo_idw_noaa;angulo_idw_bdmep;angulo_rst_noaa;
            angulo_rst_bdmep;temp_idw_noaa;temp_idw_bdmep;temp_rst_noaa;
            temp_rst_bdmep\n" % (";".join(line)).replace("\n", "")
128         output.write(head)
129     else:
130         east = float(coords[0])
131         north = float(coords[1])
132
133         # ANGULO
134         noaa = grass.raster_what('sc_estacoes_noaa_idw_rw', (east, north))
135         result_noaa = float(noaa[0]['sc_estacoes_noaa_idw_rw']['value'])
136         line.append("%.2f" % (result_noaa))

```

```

137
138     bdmap = grass.raster_what('sc_estacoes_bdmap_idw_rw', ( east, north ) )
139     result_bdmap = float(bdmap[0]['sc_estacoes_bdmap_idw_rw']['value'])
140     line.append("%.2f" % (result_bdmap))
141
142     noaa_rst = grass.raster_what('sc_estacoes_noaa_rst_rw', ( east, north ) )
143     result_noaa_rst = float(noaa_rst[0]['sc_estacoes_noaa_rst_rw']['value'])
144     line.append("%.2f" % (result_noaa_rst))
145
146     bdmap_rst = grass.raster_what('sc_estacoes_bdmap_rst_rw', ( east, north )
147     )
148     result_bdmap_rst = float(bdmap_rst[0]['sc_estacoes_bdmap_rst_rw']['value']
149     ])
150     line.append("%.2f" % (result_bdmap_rst))
151
152     #TEMPERATURA
153     noaa = grass.raster_what('sc_estacoes_noaa_idw_temp', ( east, north ) )
154     result_noaa = float(noaa[0]['sc_estacoes_noaa_idw_temp']['value'])
155     line.append("%.2f" % (result_noaa))
156
157     bdmap = grass.raster_what('sc_estacoes_bdmap_idw_temp', ( east, north ) )
158     result_bdmap = float(bdmap[0]['sc_estacoes_bdmap_idw_temp']['value'])
159     line.append("%.2f" % (result_bdmap))
160
161     noaa_rst = grass.raster_what('sc_estacoes_noaa_rst_temp', ( east, north )
162     )
163     result_noaa_rst = float(noaa_rst[0]['sc_estacoes_noaa_rst_temp']['value'])
164     line.append("%.2f" % (result_noaa_rst))
165
166     bdmap_rst = grass.raster_what('sc_estacoes_bdmap_rst_temp', ( east, north
167     )
168     )
169     result_bdmap_rst = float(bdmap_rst[0]['sc_estacoes_bdmap_rst_temp']['value
170     '])
171     line.append("%.2f" % (result_bdmap_rst))
172
173     output.write("%s\n" % (";".join(line)) )
174
175     conta_linha += 1
176
177 output.close()

```

H Script 06: Compara os valores obtidos pelo software com os observados

Nome: "06_compare.py".

```
1  #- coding: utf-8 #-
2
3  import glob, codecs, csv, os, requests
4  import sqlite3 as lite
5  import pandas as pd
6  from pylab import plt
7  import numpy as np
8  from bs4 import BeautifulSoup
9
10 def get_declination(lat, lon):
11     headers = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit
12               /537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36'}
13     with requests.Session() as s:
14         norte_sul = ("S" if lat < 0 else "N")
15         leste_oeste = ("W" if lon < 0 else "E")
16         lat = lat * -1.
17         lon = lon * -1.
18         login_page = "https://www.ngdc.noaa.gov/geomag-web/calculators/
19                     calculateDeclination?browserRequest=true&magneticComponent=d&lat1=%s&
20                     lat1Hemisphere=%s&lon1=%s&lon1Hemisphere=%s&model=WMM&startYear=2019&
21                     startMonth=10&startDay=3&resultFormat=html&ajax=true" % (
22                         lat, norte_sul, lon, leste_oeste
23                     )
24         #login
25         login = s.get(login_page, headers=headers) #, headers=headers
26         soup = BeautifulSoup(login.content, 'lxml')
27         declination = str(soup.findAll('td')[10]).split()[2].replace("Â°", "")
28
29     return -float(declination)
30
31 diretorio = os.getcwd()
32
33 output_ang_calc_noaa_idw = "%s\\comparacao\\ang_calc_noaa_idw.dat" % diretorio
34 output_ang_calc_bdmep_idw = "%s\\comparacao\\ang_calc_bdmep_idw.dat" % diretorio
35 output_ang_calc_noaa_rst = "%s\\comparacao\\ang_calc_noaa_rst.dat" % diretorio
36 output_ang_calc_bdmep_rst = "%s\\comparacao\\ang_calc_bdmep_rst.dat" % diretorio
37
38 output_noaa_idw = codecs.open(output_ang_calc_noaa_idw, "w", "utf-8")
39 output_bdmep_idw = codecs.open(output_ang_calc_bdmep_idw, "w", "utf-8")
40 output_noaa_rst = codecs.open(output_ang_calc_noaa_rst, "w", "utf-8")
41 output_bdmep_rst = codecs.open(output_ang_calc_bdmep_rst, "w", "utf-8")
42
43 output_noaa_idw.write("y\n")
44 output_bdmep_idw.write("y\n")
45 output_noaa_rst.write("y\n")
46 output_bdmep_rst.write("y\n")
```

```

45
46 arquivo_csv = diretorio + "\\aerodromos\\aerodromos_sc.csv"
47 with open(arquivo_csv, 'r') as csvfile:
48     conta = 0
49     plots = csv.reader(csvfile, delimiter=';')
50     for row in plots:
51         if conta > 0:
52             lat = float(row[4])
53             lon = float(row[5])
54             ang_observado = round(float(row[6].split("/") [0]))*10
55             declination = get_declination(lat, lon)
56
57             #9 NOAA IDW
58             ang_calc_noaa_idw = (float(row[9])+declination)
59             #10 BDMEP IDW
60             ang_calc_bdmepe_idw = (float(row[10])+declination)
61             #11 NOAA RST
62             ang_calc_noaa_rst = (float(row[11])+declination)
63             #12 BDMEP RST
64             ang_calc_bdmepe_rst = (float(row[12])+declination)
65
66             # NOAA IDW
67             if ang_observado-ang_calc_noaa_idw < -90:
68                 dif_ang_calc_noaa_idw = 180 + (ang_observado-ang_calc_noaa_idw)
69             elif ang_observado-ang_calc_noaa_idw > 90:
70                 dif_ang_calc_noaa_idw = -(180 - (ang_observado-ang_calc_noaa_idw))
71             else:
72                 dif_ang_calc_noaa_idw = ang_observado-ang_calc_noaa_idw
73
74             # BDMEP IDW
75             if ang_observado-ang_calc_bdmepe_idw < -90:
76                 dif_ang_calc_bdmepe_idw = 180 + (ang_observado-ang_calc_bdmepe_idw)
77             elif ang_observado-ang_calc_bdmepe_idw > 90:
78                 dif_ang_calc_bdmepe_idw = -(180 - (ang_observado-ang_calc_bdmepe_idw))
79             else:
80                 dif_ang_calc_bdmepe_idw = ang_observado-ang_calc_bdmepe_idw
81
82             # NOAA RST
83             if ang_observado-ang_calc_noaa_rst < -90:
84                 dif_ang_calc_noaa_rst = 180 + (ang_observado-ang_calc_noaa_rst)
85             elif ang_observado-ang_calc_noaa_rst > 90:
86                 dif_ang_calc_noaa_rst = -(180 - (ang_observado-ang_calc_noaa_rst))
87             else:
88                 dif_ang_calc_noaa_rst = ang_observado-ang_calc_noaa_rst
89
90             # BDMEP RST
91             if ang_observado-ang_calc_bdmepe_rst < -90:
92                 dif_ang_calc_bdmepe_rst = 180 + (ang_observado-ang_calc_bdmepe_rst)
93             elif ang_observado-ang_calc_bdmepe_rst > 90:
94                 dif_ang_calc_bdmepe_rst = -(180 - (ang_observado-ang_calc_bdmepe_rst))
95             else:
96                 dif_ang_calc_bdmepe_rst = ang_observado-ang_calc_bdmepe_rst
97
98             output_noaa_idw.write("%.02f\n" % (dif_ang_calc_noaa_idw))

```



```
99         output_bdmp_idw.write("%.02f\n" % (dif_ang_calc_bdmp_idw))
100         output_noaa_rst.write("%.02f\n" % (dif_ang_calc_noaa_rst))
101         output_bdmp_rst.write("%.02f\n" % (dif_ang_calc_bdmp_rst))
102
103     else:
104         print(row)
105
106     conta += 1
107
108 output_noaa_idw.close()
109 output_bdmp_idw.close()
110 output_noaa_rst.close()
111 output_bdmp_rst.close()
```