



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Ministério da
**Ciência, Tecnologia
e Inovação**



ESTUDO E DESENVOLVIMENTO DE ALGORITIMOS DE TRATAMENTO DE DADOS APLICADOS AO LEVANTAMENTO DE RECURSO SOLAR

Artur de Almeida Leme

INPE
São José dos Campos
2018

ESTUDO E DESENVOLVIMENTO DE ALGORITMOS DE TRATAMENTO DE DADOS APLICADOS AO LEVANTAMENTO DE RECURSO SOLAR

RELATÓRIO FINAL DE PROJETO DE INICIAÇÃO CIENTÍFICA (PIBIC/CNPq/INPE)

Artur de Almeida Leme (Universidade Federal de São Paulo, Bolsista
PIBIC/CNPq)

E-mail: arturleme2@gmail.com

Dr. Enio Bueno Pereira (INPE, Orientador)

E-mail: enio.pereira@inpe.br

Dr. Fernando Ramos Martins (UNIFESP/INPE, Co-Orientador)

E-mail: fernando.martins@unifesp.br

Julho de 2018

RESUMO

O avanço do desenvolvimento econômico e o estilo de vida moderno têm provocado um aumento na demanda energética. Junto a isso, as preocupações ambientais decorrentes do crescimento das emissões de poluentes atmosféricos e o cenário de vulnerabilidade do recurso hídrico no país, provocado pelas oscilações climáticas, têm apresentado importantes desafios científicos no âmbito da inovação e no desenvolvimento de alternativas para produção de energia através de fontes renováveis e de baixo impacto ambiental. Nesse contexto, a energia solar emerge com destaque. Para uma avaliação consistente do potencial e viabilidade econômica do aproveitamento solar, estudos são necessários não só para quantificar a disponibilidade, mas também a variabilidade deste recurso, tanto em escala espacial quanto temporal. Concomitante a sua crescente participação na matriz energética do país, estudos têm apontado desafios a serem superados, como exemplo, as inconsistências nos dados gerados pelos diferentes sistemas observacionais disponíveis. Desta forma, o objetivo deste trabalho é o estudo dos métodos de qualificação aplicados aos dados meteorológicos utilizados na quantificação do recurso solar nacional; mais especificamente, a terceira etapa do processo de qualificação de dados da rede SONDA, operada pelo Instituto Nacional de Pesquisas Espaciais. Três estações em específico são analisadas, de modo a tentar integrar diferentes climas e condições físicas encontradas no território brasileiro: estação de Petrolina, localizada na região Nordeste, que apresenta clima semiárido quente; estação de Brasília, localizada na região Centro-Oeste, que apresenta clima tropical com estação seca; e Estação de São Martinho da Serra, localizada na região Sul, que apresenta clima subtropical úmido. A partir dessa análise, é feita a revisão dos algoritmos de qualificação de dados e são propostas melhorias ao sistema de coleta de dados, de modo a aumentar a confiabilidade dos dados coletados e subsidiar a validação de modelos computacionais de levantamento do recurso solar no futuro.

LISTA DE FIGURAS

Figura 1 - (a) Matriz energética brasileira e (b) Matriz de energia elétrica brasileira.	2
Figura 2 - Espectro de radiação eletromagnética emitida pelo Sol e pela Terra.	5
Figura 3 - Diagrama simbólico dos processos de interação da radiação solar com a atmosfera terrestre.	6
Figura 4 - (a) Piranômetros sem e com a banda de sombreamento e (b) Pireliômetro da Estação de Cachoeira Paulista. Fonte: INPE (2016).	8
Figura 5 - Fluxograma de execução do software de validação da rede SONDA V.3.5. .	10
Figura 6 - Mapa do Brasil com as três estações de referência da rede SONDA utilizadas neste estudo.	12
Figura 7. Histogramas da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de São Martinho da Serra, por ano.	15
Figura 8. Visão geral da série temporal de dados para a estação de São Martinho da Serra.	16
Figura 9. Histogramas da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Petrolina, por ano.	17
Figura 10. Visão geral da série temporal de dados para a estação de São Martinho da Serra.	18
Figura 11. Histogramas da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Brasília, por ano.	19
Figura 12. Visão geral da série temporal de dados para a estação de Brasília.	20
Figura 13. Exemplo de inconsistência nas medições do pireliômetro.	21
Figura 14. Exemplo de inconsistência nas medições do piranômetro.	21
Figura 15. Exemplo de inconsistência nas medições de ambos os equipamentos.	22
Figura 16. Exemplo de desvio constante das faixas de aceitação.	23

LISTA DE TABELAS

Tabela 1 - Parâmetros de códigos de avaliação objetiva dos dados.	10
Tabela 2. Critérios utilizados para validação de dados radiométricos de Radiação Global Horizontal, Direta Normal e Difusa, segundo critérios das organizações BSRN e webmet.com.	11
Tabela 3. Diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de São Martinho da Serra, por ano.	14
Tabela 4. Diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Petrolina, por ano.	16
Tabela 5. Diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Brasília, por ano.	18
Tabela 6. Soma das porcentagens e média geral para cada limite, por ano, para a estação de São Martinho da Serra.	25
Tabela 7. Soma das porcentagens e média geral para cada limite, por ano, para a estação de Brasília.	25
Tabela 8. Soma das porcentagens e média geral para cada limite, por ano, para a estação de Petrolina.	25

SUMÁRIO

1 INTRODUÇÃO	1
2 FUNDAMENTAÇÃO TEÓRICA	3
2.1 Energia solar	3
2.2 Balanço radioativo da atmosfera e seus efeitos	4
2.3 Equipamentos de medição da radiação solar e seu funcionamento	6
2.4 Rede SONDA	8
2.5 Qualificação de dados	9
2.5.1 Processo	9
2.5.2 Código	10
2.5.3 Critérios	10
3 METODOLOGIA	11
4 DISCUSSÃO DE RESULTADOS	13
4.1 Panorama Geral.....	13
4.1.1 Estação de São Martinho da Serra	13
4.1.2 Estação de Petrolina	16
4.1.3 Estação de Brasília.....	18
4.2 Análise das inconsistências encontradas.....	20
4.2.1 Inconsistência nas medições do pireliômetro.....	20
4.2.2 Inconsistência nas medições do piranômetro.....	21
4.2.3 Inconsistência nas medições de ambos os equipamentos.....	22
4.2.4 Desvio constante das faixas de aceitação.....	22
4.3 Deficiências encontradas e melhorias propostas.....	23
4.3.1 Problemas operacionais com os sensores.....	23
4.3.2 Falta de manutenção nos equipamentos.....	24
4.3.3 Análise dos limites utilizados nos critérios de qualificação de dados	25
5 CONSIDERAÇÕES FINAIS	26

REFERÊNCIAS BIBLIOGRÁFICAS	27
APÊNDICE A – CÓDIGO DE ELABORAÇÃO DOS GRÁFICOS	28
APÊNDICE B – CÓDIGO DE ELABORAÇÃO DOS HISTOGRAMAS	138

1. INTRODUÇÃO

A relação entre os estágios de desenvolvimento humano e consumo de energia demonstra como a evolução no uso dos recursos energéticos propiciou o desenvolvimento da sociedade humana ao longo do tempo, desde a Idade Antiga até a sociedade altamente tecnológica (GOLDEMBERG e VILLANUEVA, 2003). A questão energética é de suma importância no desenvolvimento da sociedade e da economia, basta analisar o progresso das técnicas de trabalho em diferentes áreas das atividades humanas, como a indústria, alimentação, agricultura, moradia, comércio e locomoção, que foram proporcionadas através do consumo e uso de recursos energéticos.

O aumento da demanda e consumo de energia decorrente do progresso tecnológico e do avanço no desenvolvimento humano, são apontados como os principais fatores na aceleração das alterações climáticas e ambientais (MARTINS e PEREIRA, 2007). O uso de recursos fósseis como o petróleo e o carvão, representam aproximadamente, 79% da matriz de consumo final mundial (KUMAR et al., 2010). A queima desses combustíveis fósseis, causam grandes quantidades de emissões de gases poluentes atmosféricos, principalmente do efeito estufa (GEE), gerando grandes alterações climáticas e ambientais, como o fenômeno de aquecimento global.

A sociedade, juntamente com a comunidade científica e os governos, têm procurado opções de fontes energéticas limpas e de baixo impacto ambiental, motivados não só pela busca de meios que venham a complementar a atual demanda de energia e seus cenários de crescimento, como também pelo apelo da diminuição do uso de combustíveis fósseis, em função dos problemas ambientais e do possível esgotamento destes recursos nas próximas décadas (BENTLEY, 2002 e GELLER, 2003).

Neste cenário atual, onde é considerado como certo a continuação do aumento da demanda energética mundial, em contrapartida, uma diminuição no uso de combustíveis fósseis, as fontes renováveis de energia com baixa emissão de carbono, vem ganhando um papel fundamental nesta discussão, gerando acordos internacionais e previsões de grandes investimentos para os próximos anos. A tendência geral é que os preços das tecnologias para aproveitamento de fontes renováveis de energia venham a diminuir com o aumento dos estudos e aprimoramento das técnicas já conhecidas. O Brasil já possui uma matriz energética e uma matriz elétrica com participação ampla de fontes renováveis, principalmente o recurso hidroelétrico, conforme mostra a Figura 1.

Neste contexto, se destaca o uso da energia solar, que apresenta diversos benefícios, como a viabilização de atendimento de regiões remotas onde o custo é demasiadamente elevado

em termos de sistemas convencionais, regulação da oferta de energia em períodos de estiagem, além de permitir a diminuição do uso de combustíveis fósseis para a geração de eletricidade.

O Brasil, por estar localizado em sua maior parte na região tropical, possui grande potencial de energia solar durante todo o ano. Pode se dizer que a utilização de energia solar traria um gama de benefícios a longo prazo, principalmente em regiões remotas onde o custo da eletricidade pela rede convencional é extremamente alta, viabilizando o desenvolvimento socioeconômico destas áreas. Esta fonte energética apresenta elevado potencial de crescimento no Brasil, contudo seu crescimento comercial e investimentos em larga escala vem se esbarrando na escassez de informações sobre a disponibilidade e viabilidade relacionadas a esse recurso. A maior parte dessas informações provem do setor de pesquisa básica e climatológica, sendo constituída por séries incompletas e dispersas, pois essas informações geralmente foram coletadas para outros fins que não aquele do setor energético, portanto constituem uma base de dados inadequadas e insuficientes para esse fim (MARTINS et al., 2008).

Para que sejam implementados e desenvolvidos os sistemas de conversão de energia solar no Brasil, tanto autônomas quanto nas grandes centrais, é muito importante o mapeamento do potencial solar no território brasileiro, de modo a fomentar a migração para uma matriz energética limpa e menos dependente dos recursos hídricos e fósseis. Estudos para o mapeamento do potencial de energia solar podem ser desenvolvidos por meio de diferentes métodos. As principais metodologias são a utilização de uma rede de radiômetros, que devem estar distribuídos de forma estratégica sobre uma região de interesse; ou o uso de modelos computacionais para parametrização dos processos radiativos e solução da equação da transferência radiativa na atmosfera.

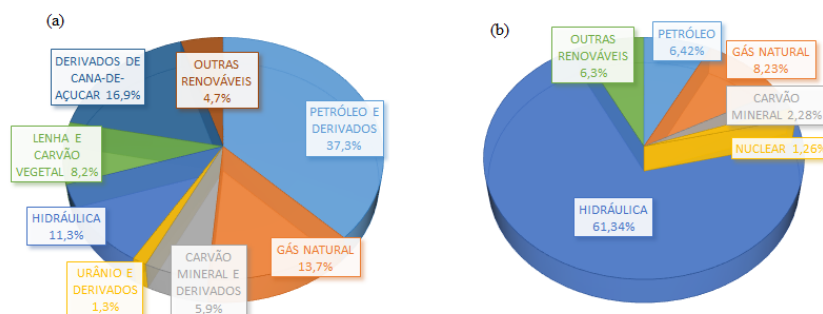


Figura 1 - (a) Matriz energética brasileira e (b) Matriz de energia elétrica brasileira. Fonte: ANEEL (2016).

É muito importante que para uma avaliação consistente destes mencionados potenciais, sejam realizados estudos que não só quantifiquem sua disponibilidade, mas também a sua variabilidade, em escala tanto espacial quanto temporal. Sendo assim, o Instituto Nacional de Pesquisas Espaciais, no âmbito das atividades desenvolvidas dentro do Centro de Ciência do Sistema Terrestre, possui competência em estudos de fontes renováveis de energia através do LABREN – Laboratório de Modelagem e Estudos de Recursos Renováveis de Energia. Este grupo realiza de pesquisa em meteorologia aplicada ao setor de energia e sobre seus impactos no sistema climático, com foco na observação, na modelagem computacional e no desenvolvimento de cenários face as mudanças ambientais globais. Para quantificar o recurso solar, deve-se valer de dados observacionais, que envolvem variáveis meteorológicas medidas em diversas estações. Por conta disso, em função do elevado volume de dados gerados pelos diferentes sistemas observacionais disponíveis, é comum estabelecer rotinas computacionais para verificar e certificar a qualidade da base de dados, sinalizando possíveis inconsistências.

Este trabalho teve como objetivo geral o estudo dos métodos de tratamento e qualificação de dados meteorológicos e ambientais, utilizados na quantificação do recurso solar nacional. Foram revisados os critérios de qualificação atuais, ou seja, foi compreendido os critérios de qualificação hoje aplicados aos dados solarimétricos e aos dados espaciais, verificando sua consistência e analisando possíveis falhas. Foram realizadas também, avaliações dos pontos críticos e formulação de novos critérios, ou seja, foi realizado uma análise das deficiências encontradas e proposto alterações nos critérios que apresentam problemas.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 Energia solar

O Sol é a maior fonte de energia do sistema Terra-Atmosfera, possuindo um diâmetro de $1,39 \times 10^6$ km e um volume de $1,412 \times 10^{18}$ km³. A energia proveniente do Sol que chega até a superfície terrestre, propagando-se como energia radiante, é denominada de radiação solar. A energia solar é produzida através de reações químicas e nucleares (ou fusão nuclear), que consiste basicamente na reação de dois átomos de hidrogênio formando um átomo de hélio. A radiação solar apresenta um papel fundamental por constituir na principal força motriz para processos térmicos, dinâmicos e químicos da Terra, como a circulação geral da atmosfera.

A quantidade de energia radiante, num intervalo unitário de comprimento de onda, que atravessa uma unidade de área tomada perpendicularmente a direção considerada, na unidade de tempo, é a radiância, sendo a principal característica de um campo de radiação. A constante solar, é a irradiância solar que atinge o topo da atmosfera terrestre, apresenta o valor aproximado de 1367 W/m^2 . Esse parâmetro apresenta variações ao longo do ano, devido aos efeitos astronômicos, principalmente aqueles ligados a órbita da Terra ao redor do Sol. A irradiância solar que incide na superfície da Terra depende de alguns fatores, como o comprimento de onda, de fenômenos que ocorrem na própria fonte de energia (erupções solares, manchas solares, variações de temperatura na atmosfera solar), da distância do Sol-Terra (associado ao movimento de rotação da Terra e à órbita da Terra em torno do Sol), além de outros fatores associados a interações da radiação eletromagnética com os constituintes da atmosfera da Terra (como as nuvens, umidade, ozônio e aerossóis), que alteram a disponibilidade de energia solar na superfície em faixas específicas de comprimentos de onda. A razão entre a irradiância solar incidente na superfície terrestre e no topo da atmosfera, é definida como transmitância atmosférica (IQBAL, 2012).

2.2 Balanço radioativo da atmosfera e seus efeitos

A radiação eletromagnética solar, normalmente nomeada como “radiação de ondas curtas”, apresenta comprimentos de ondas entre 0 e 4 micrometros. Parte dessa radiação incide na superfície terrestre é absorvida e reemitida para o espaço. A radiação que é reemitida para o espaço é conhecida como radiação terrestre, normalmente nomeadas como “radiação de ondas longas”, por apresentarem comprimentos de onda superiores a 4 micrometros. A Figura 2 apresenta o espectro da radiação solar e da radiação terrestre de forma a facilitar a compreensão dos termos de radiação de ondas curtas e ondas longas, respectivamente.

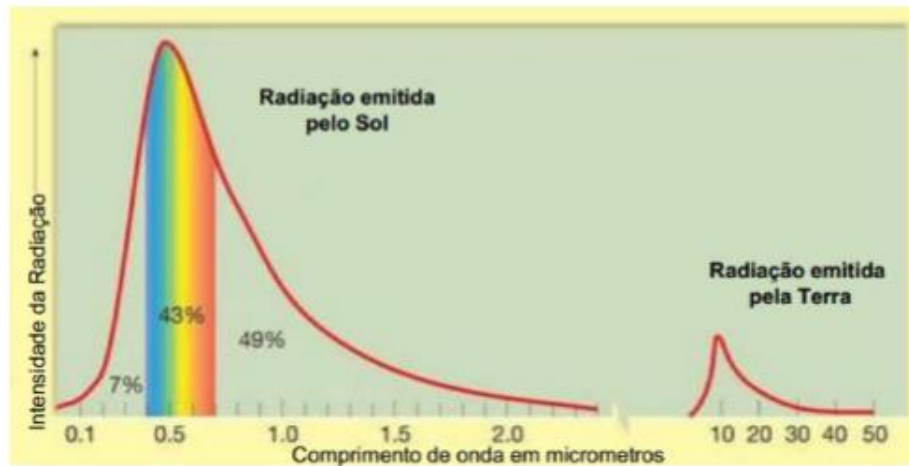


Figura 2 - Espectro de radiação eletromagnética emitida pelo Sol e pela Terra. Fonte: Adaptado de IQBAL (2012).

Ao atravessar a atmosfera, há três tipos de incidência de energia na superfície terrestre: a radiação direta presente no feixe de energia emitida pelo Sol e que não sofre processos de interação com os constituintes atmosféricos; a radiação difusa, que devido as interferências de fatores atmosféricos, faz com que parte da energia seja espalhada em diversas direções; e por último, a radiação global, que é o conjunto da radiação direta e radiação difusa.

A radiação eletromagnética é atenuada por processos de espalhamento e absorção pelas partículas (denominados aerossóis) e moléculas de gases atmosféricos. Os coeficientes de absorção e de espalhamento dependem do comprimento de onda da radiação, além das espécies de gases e aerossóis presentes na atmosfera.

Aproximadamente 30% da radiação incidente no topo da atmosfera são refletidos de volta para o espaço pelas nuvens, pela superfície do planeta, e pelos gases e partículas atmosféricas. Os outros 70% são absorvidos produzindo aquecimento do sistema e causando evaporação de água (calor latente) ou convecção (calor sensível). A Figura 3 apresenta um diagrama simplificado para representação dos processos radiativos na atmosfera.

O processo de absorção da energia solar por constituintes atmosféricos, alteram o espectro eletromagnético da radiação solar incidente, à medida que a mesma percorre a atmosfera terrestre, atenuando principalmente a radiação eletromagnética na região da radiação ultravioleta. A radiação nesta faixa de comprimento de onda é intensamente absorvida na camada de Ozônio (situada cerca de 35 km acima da superfície) em razão das reações fotoquímicas que ocorrem nessa camada da atmosfera. O processo de absorção da

radiação solar é seletiva, destacando o ozônio (O₃), o vapor d' água e o dióxido de carbono (CO₂) como os principais agentes absorvedores, sendo os dois últimos os principais gases do efeito estufa. A energia absorvida pelo sistema Terra-Atmosfera é reemitida na faixa do infravermelho do espectro de radiação eletromagnética, sendo que aproximadamente 6% são provenientes da superfície e aproximadamente 64% tem origem em nuvens e constituintes atmosféricos.

O processo de espalhamento é um fenômeno natural no qual a radiação incidente interage com os constituintes atmosféricos de forma a alterar a direção de propagação das ondas eletromagnéticas. Destacam-se dois processos de espalhamentos: Mie, onde partículas grandes (com tamanhos próximos ao do comprimento de onda da radiação) são responsáveis pela tonalidade branca, exibida por nuvens, nevoeiros e a abundância de particulados na atmosfera de núcleos urbanos; e Rayleigh, onde partículas pequenas (com dimensões muito menores que o comprimento de onda da radiação) são responsáveis pela tonalidade azul do céu durante o dia (IQBAL, 2012).

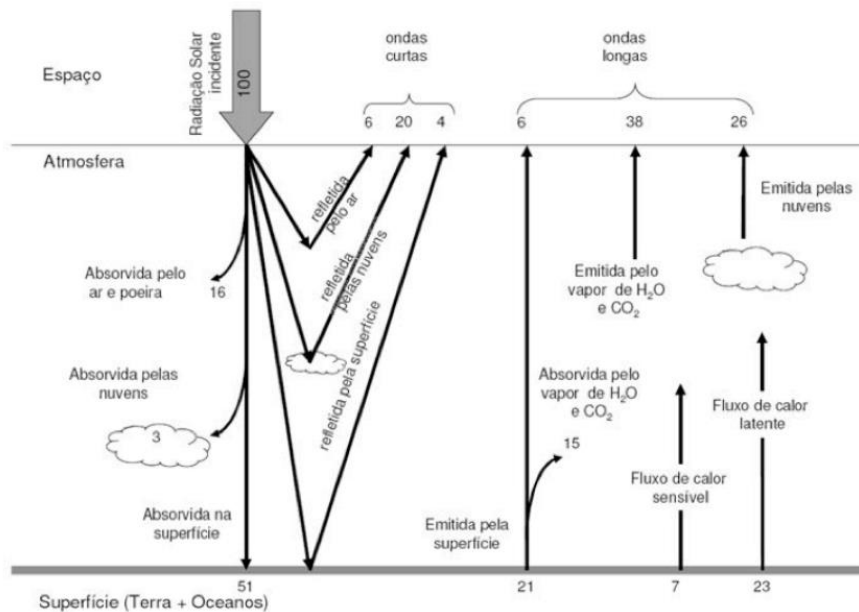


Figura 3 - Diagrama simbólico dos processos de interação da radiação solar com a atmosfera terrestre.

Fonte: PEREIRA et al.(2014).

2.3 Equipamentos de medição da radiação solar e seu funcionamento

Os dois principais equipamentos utilizados na medição da radiação solar, tanto da global como das componentes direta e difusa, na superfície terrestre são o piranômetro e o pireliômetro, o primeiro para medidas de irradiação global e difusa; e o segundo para medidas da irradiação direta.

O piranômetro se diferencia em dois tipos: o piranômetro termoelétrico e o piranometro fotovoltaivo. O primeiro, utilizado para medir a irradiância solar global, normalmente no plano horizontal, tem como sensor uma termopilha colocada no interior de duas semiesferas de vidro concêntricas. A termopilha é construída com múltiplos termopares em série, que medem a diferença de temperatura entre as duas superfícies, uma pintada de preto e outra de branco igualmente iluminadas. A expansão sofrida pelas superfícies provoca um diferencial de potencial que, ao ser medida, mostra o valor instantâneo da energia solar. Já o segundo, utiliza uma célula fotovoltaica de silício monocristalino para coletar medidas solarimétricas. Os piranômetros fotovoltaicos tem como vantagem apresentar um custo bem menor, contudo tem como desvantagem o fornecimento de medidas com menor precisão.

O pireliômetro é um instrumento utilizado para medir a irradiância direta com incidência normal a superfície. A irradiância difusa é bloqueada instalando-se o sensor termoelétrico dentro de um tubo de colimação, com paredes enegrecidas e apontado diretamente ao Sol (dispositivo de rastreamento). O instrumento caracteriza-se por apresentar uma pequena abertura de forma a “visualizar” apenas o disco solar e a região vizinha denominada circunsolar. O instrumento precisa estar acoplado a um sistema de rastreamento do movimento solar para que esteja constantemente ajustado para a direção do Sol.

O valor da irradiação difusa é medido com um piranômetro ao qual é acoplado uma banda ou um disco de sombreamento para bloquear a componente direta. Ao combinar os valores medidos pelo piranômetro bloqueado com o outro sem o dispositivo de bloqueio pode-se obter a componente direta pela subtração dos valores da irradiância global e da sua componente difusa. A banda de sombreamento necessita de ajuste manual de acordo com a declinação solar, diferentemente do disco de sombreamento, que é utilizado em conjunto com um rastreador solar. Em ambos os casos são necessárias correções das medições, que envolvam a compensação do sombreamento do céu causado pelo anel ou disco (PEREIRA et al., 2014).

A Figura 4 apresenta os principais equipamentos utilizados na medição da radiação solar: (a) Piranômetros sem a banda de sombreamento e com a banda de sombreamento e (b) Pireliômetro, da estação localizada em Cachoeira Paulista.



Figura 4 - (a) Piranômetros sem e com a banda de sombreamento e (b) Pireliômetro da Estação de Cachoeira Paulista. Fonte: INPE (2018).

Os instrumentos de medição da radiação solar global, direta e difusa são classificados conforme a sua precisão. De acordo com a norma ISO 9060, os instrumentos são classificados em três categorias: padrão secundário, primeira classe e segunda classe. Segundo a WMO, os instrumentos podem ser classificados em alta qualidade (erro máximo de 2%), boa qualidade (5%) e qualidade razoável (10%).

2.4 Rede SONDA

A rede SONDA (Sistema de Organização Nacional de Dados Ambientais) é um projeto executado pelo Instituto Nacional de Pesquisas Espaciais (INPE) para a implementação de infraestrutura física e de recursos humanos destinados a instalar, operar e disponibilizar uma base de dados de superfície que atenda a demanda de informações sobre recursos de energia solar e eólica no Brasil.

Atualmente, a rede SONDA conta com um total de 18 estações distribuídas em todo território nacional, para que assim possa abranger todos os climas típicos do Brasil. Das 18 estações, 13 são próprias e as outras 5 são de empresas e instituições parceiras.

As estações coletam dados meteorológicos básicos, como temperatura, pressão atmosférica, umidade e cobertura de nuvens. As estações solarimétricas, também coletam dados relacionadas a energia solar, como as componentes global, direta e difusa do fluxo de radiação solar incidente. Já as estações anemométricas, coletam dados relacionadas a caracterização dos ventos, como direção e velocidade do vento em 10m, 25m e 50m de altura. Como este trabalho é direcionado a energia solar, o foco deste artigo é dado apenas nas estações solarimétricas.

Todos os instrumentos de coleta de dados utilizados na rede SONDA são de primeira classe, de modo a atender os padrões estabelecidos pela Organização Mundial de Meteorologia (OMM) para a rede internacional BSRN (Baseline Surface Radiation Network). Todos os piranômetros e pirheliômetros são manufaturados por Eppley Lab. ou Kipp&Zonen. Quatro estações da rede SONDA, integram a BSRN e atendem todos os padrões de qualidade exigidos para a instrumentação instalada, operação/manutenção e controle de qualidade de dados. As demais estações não possuem instrumentação para aquisição de dados de irradiação solar direta e, por essa razão, não estão integradas a BSRN (INPE, 2018).

2.5 Qualificação de dados

2.5.1 Processo

O processo de qualificação dos dados solarimétricos e espaciais, visa oferecer dados com maior confiabilidade para acesso e uso público e gratuito. Esse processo, consiste de quatro níveis sequenciais, desde filtros mais grosseiros até filtros mais refinados. Esses filtros sinalizam quando um dado é considerado suspeito de incorreção, através da execução de algoritmos que adotam alguns critérios aceitos internacionalmente bastante rigorosos com base na experiência nos testes de qualidade realizados da Baseline Solar Radiation Network (BSRN) e outros desenvolvidos pela experiência do grupo de Energias Renováveis do Centro de Ciências do Sistema Terrestre (CCST).

A aprovação em cada etapa é requisito para a continuidade do processo. Logo, quando um dado é considerado aprovado numa etapa, o dado receberá o código de aprovado e a etapa seguinte será iniciada. Contudo, se o dado for considerado com qualidade suspeita, o processo será interrompido e o dado receberá o código equivalente a suspeito. No primeiro nível, são testados os limites fisicamente possíveis, onde são esperados valores dentro de um intervalo que atende os critérios estabelecidos por modelos teóricos e leis da física. No segundo nível, são filtrados os eventos extremamente raros, que podem ocorrer com frequência baixa em decorrência de condições atmosféricas bastante específicas como a geometria, Sol-Nebulosidade-Superfície. Já no terceiro nível, as variáveis são comparadas com outras variáveis coletadas na mesma estação, de modo a verificar a consistência do conjunto de dados observados na estação. Nesta etapa, o dado é suspeito se seu valor apresente uma inconsistência com valores observados para outras variáveis. No quarto e último nível, a radiação global, direta e difusa, são comparadas com o modelo numérico de transferência radiativa utilizado para estimar a irradiância solar em

condições de céu claro (INPE, 2018). O modelo adotado nesta etapa é conhecido como Parametrization Model C (Iqbal, 1983). O dado é considerado suspeito caso a medida esteja inconsistente quando comparada com estimativas do modelo computacionais. Uma visão geral do processo é dado pelo fluxograma da Figura 5.

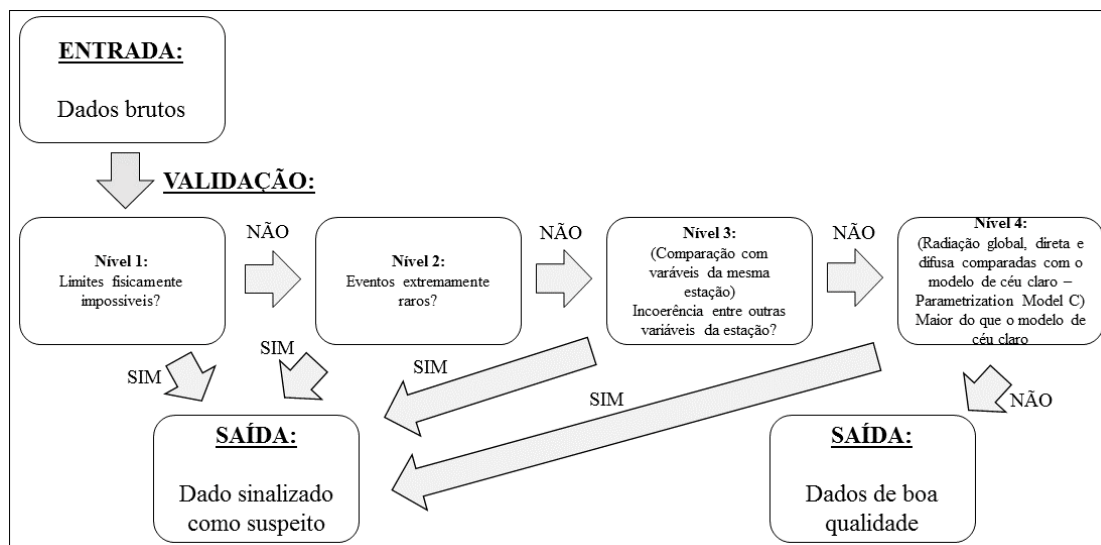


Figura 5 - Fluxograma de execução do software de validação da rede SONDA V.3.5.

2.5.2 Código

Cada dado terá um qualificador de quatro dígitos, onde o resultado de cada uma das etapas aplicadas no controle de qualidade é dado na forma de códigos numéricos, lidos da direita para a esquerda, que classificam o valor medido dentre dos parâmetros apresentado na Tabela 1.

0	Nenhum procedimento foi executado.
2	Dado suspeito de ser incorreto.
5	Procedimento não pode ser executado.
9	Dado de boa qualidade ou não suspeito.

Tabela 1 - Parâmetros de códigos de avaliação objetiva dos dados. Fonte: INPE (2018).

2.5.3 Critérios

Segundo os limites baseados em critérios das organizações BSRN e webmet.com, os critérios utilizados para validação de dados radiométricos de Radiação Global Horizontal, Direta Normal e Difusa utilizados nos algoritmos são mostrados na Tabela 2. A partir dos critérios mostrados na tabela os dados podem ser classificados dentre os níveis sequenciais mostrados anteriormente.

<i>Fisicamente Possível (algoritmo 1)</i>	<i>Extremamente Raro (algoritmo 2)</i>	<i>Comparação entre Sensores (algoritmo 3)</i>
Global Horizontal (Wm^{-2}) variável: glo_avg		
1. desvio padrão $\neq 0$ (zero)	min.: -2	(Global) / (Sum) deve estar entre $\pm 10\%$ de 1,0 para $AZS < 75^\circ$, $Sum > 50$
2. min.: -4 max.: $S_a \times 1,5 \times \mu_0^{1,2} + 100$	max.: $S_a \times 1,2 \times \mu_0^{1,2} + 50$	(Global) / (Sum) deve estar entre $\pm 15\%$ de 1,0 para $93^\circ > AZS > 75^\circ$, $Sum > 50$
		Obs.: este teste não é possível para $Sum < 50$
Direta Normal (Wm^{-2}) variável: dir_avg		
1. desvio padrão $\neq 0$ (zero)		
2. verificação do estado de operação do rastreador solar (onde existente)	min.: -2	Direta $\times \mu_0 - 50 \leq$ direta horizontal (Global - Difusa) \leq Direta $\times \mu_0 + 50$
3. min.: -4 max.: S_a	max.: $S_a \times 0,95 \times \mu_0^{0,2} + 10$	
Difusa (Wm^{-2}) variável: diff_avg		
1. desvio padrão $\neq 0$ (zero)		(Difusa) / (Global) $< 1,05$ para $AZS < 75^\circ$, Global > 50
2. verificação do estado de operação do rastreador solar (onde existente)	min.: -2	(Difusa) / (Global) $< 1,10$ para $93^\circ > AZS > 75^\circ$, Global > 50
3. min.: -4 max.: $S_a \times 0,95 \times \mu_0^{1,2} + 50$	max.: $S_a \times 0,75 \times \mu_0^{1,2} + 30$	Obs.: este teste não é possível para Global < 50

Tabela 2. Critérios utilizados para validação de dados radiométricos de Radiação Global Horizontal, Direta Normal e Difusa, segundo critérios das organizações BSRN e webmet.com. Fonte: INPE (2018).

3. METODOLOGIA

Neste trabalho, foram analisados os algoritmos e a terceira etapa do processo de qualificação de dados de Radiação Direta Normal de 3 estações solarimétricas integradas a rede BSRN. As estações analisadas neste estudo, foram as estações de referência: a Estação de São Martinho da Serra, localizada no estado do Rio Grande do Sul; a Estação de Brasília, localizada no Distrito Federal no Centro-oeste brasileiro; e a Estação de Petrolina, que se localiza no interior do estado de Pernambuco, região Nordeste do Brasil, respectivamente. A Figura 6 apresenta a localização geográfica das estações avaliadas.



Figura 6 - Mapa do Brasil com as três estações de referência da rede SONDA utilizadas neste estudo.

Uma base de dez anos de dados foi utilizada para o estudo. Essa série temporal compreendeu de janeiro de 2008 a setembro de 2017 para a Estação de São Martinho da Serra, de fevereiro de 2005 a novembro de 2015 para a Estação de Brasília, e de janeiro de 2008 a setembro de 2017 para a Estação de Petrolina.

A partir dos dados minuto a minuto das estações citadas anteriormente obtidos no website da rede SONDA com valores de dia juliano, minuto do dia, Radiação Global, Radiação Direta e Radiação Difusa, a implementação das análises foi realizada com o desenvolvimento de rotinas computacionais em linguagem Python. Através dessa linguagem, foi desenvolvido o código computacional para comparação entre os valores observados da Radiação Direta Normal à superfície com uso de pirheliometro e valores indiretos de mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa, através de gráficos de dispersão mensal (um exemplo do código é dado no Apêndice A). Essa comparação foi feita a partir da Equação 1, que compreende a comparação entre sensores da mesma estação, segundo o algoritmo 3 do processo de qualificação de dados, onde μ_0 é o cosseno do ângulo zenital.

$$\text{Máximo: (Direta Normal} \cdot \mu - 50) \leq \text{Direta Horizontal (Global-Difusa)} \leq (\text{Direta Normal} \cdot \mu + 50) \quad (1)$$

Além disso, foram elaborados histogramas para análise de frequências da diferença dada entre valores observados da Radiação Direta Normal à superfície e valores indiretos de mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa, também com base na equação 1 mostrada anteriormente, a partir de rotinas computacionais em linguagem Python (um exemplo do código é dado no Apêndice B). Em linhas gerais, os produtos deste trabalho permitiram investigar a consistência da base de dados coletados utilizando diferentes sensores. A partir dos gráficos plotados, foram analisados e discutidos os processos de qualificação de dados e os limites e critérios de aceitação utilizados, verificando sua consistência e apontando falhas e deficiências encontradas.

4. DISCUSSÃO DE RESULTADOS

4.1 Panorama Geral

Para cada estação analisada, foi realizada uma análise de frequência de dados por ano em função da diferença entre valores observados da Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa (segundo a Equação 1), e plotados histogramas para melhor visualização dos resultados. Essa seção abordará um panorama geral dos gráficos plotados.

4.1.1 Estação de São Martinho da Serra

A estação de São Martinho da Serra foi a que obteve a maior quantidade de dados aprovados pela terceira etapa do processo de qualificação, tendo uma média geral de aprovação de 75,34%. Na Tabela 3 e nas demais tabelas desta seção, são mostrados os valores da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa. Desse modo, de acordo a Equação 1, a faixa de aceitação por esta etapa da qualificação é dada por valores positivos de 50 e valores negativos de 50 (de +50 a -50).

São Martinho da Serra											
Diferença	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	Média
-200 : -176	0,269%	1,127%	0,952%	1,203%	1,974%	1,366%	0,033%	0,014%	0,036%	0,005%	0,698%
-175 : -151	0,411%	1,728%	1,200%	1,439%	2,291%	1,770%	0,127%	0,016%	0,058%	0,005%	0,905%
-150 : -126	0,566%	2,064%	1,874%	2,529%	3,899%	2,494%	0,330%	0,389%	0,357%	0,005%	1,451%
-125 : -101	1,123%	2,263%	2,010%	2,441%	3,907%	2,284%	0,790%	0,874%	0,698%	0,007%	1,640%
-100 : -76	2,058%	3,444%	2,848%	2,891%	3,753%	3,000%	2,049%	1,011%	0,975%	0,008%	2,204%
-75 : -51	3,487%	4,952%	4,203%	4,503%	5,106%	4,412%	3,911%	1,484%	1,414%	0,053%	3,352%
-50 : -26	6,063%	6,477%	6,404%	6,604%	6,811%	6,836%	6,989%	2,297%	2,363%	0,253%	5,110%
-25 : 0	12,797%	13,167%	13,664%	12,422%	12,554%	12,838%	13,040%	32,873%	30,825%	45,502%	19,968%
0 : 25	48,683%	47,730%	48,146%	48,016%	47,316%	48,186%	48,956%	50,468%	51,552%	51,362%	49,042%
26 : 50	1,193%	0,814%	1,044%	0,760%	0,580%	0,774%	1,219%	2,639%	3,173%	0,035%	1,223%
51 : 75	1,200%	0,690%	0,937%	0,775%	0,566%	0,739%	1,159%	2,305%	2,616%	0,009%	1,100%
76 : 100	1,218%	0,696%	0,938%	0,855%	0,525%	0,662%	1,112%	1,848%	2,183%	0,009%	1,005%
101 : 125	1,150%	0,637%	0,811%	0,875%	0,501%	0,685%	1,109%	1,644%	1,359%	0,009%	0,878%
126 : 150	1,090%	0,543%	0,779%	0,803%	0,428%	0,602%	1,157%	0,865%	0,856%	0,009%	0,713%
151 : 175	1,097%	0,546%	0,741%	0,834%	0,417%	0,630%	1,072%	0,117%	0,086%	0,009%	0,555%
176 : 200	1,051%	0,506%	0,732%	0,780%	0,380%	0,632%	1,098%	0,022%	0,002%	0,009%	0,521%

Tabela 3. Diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de São Martinho da Serra, por ano.

Pode se observar que a maior das médias das diferenças para a estação de São Martinho da Serra foi entre os valores de 0 a 25 (49,042%), e a menor entre os valores de 176 a 200 (0,521%).

Os histogramas plotados para esta estação são mostrados na Figura 7. Pode se observar claramente uma tendência para a parte inferior dos limites de aceitação (maior frequência de dados tendendo ao lado esquerdo do gráfico).

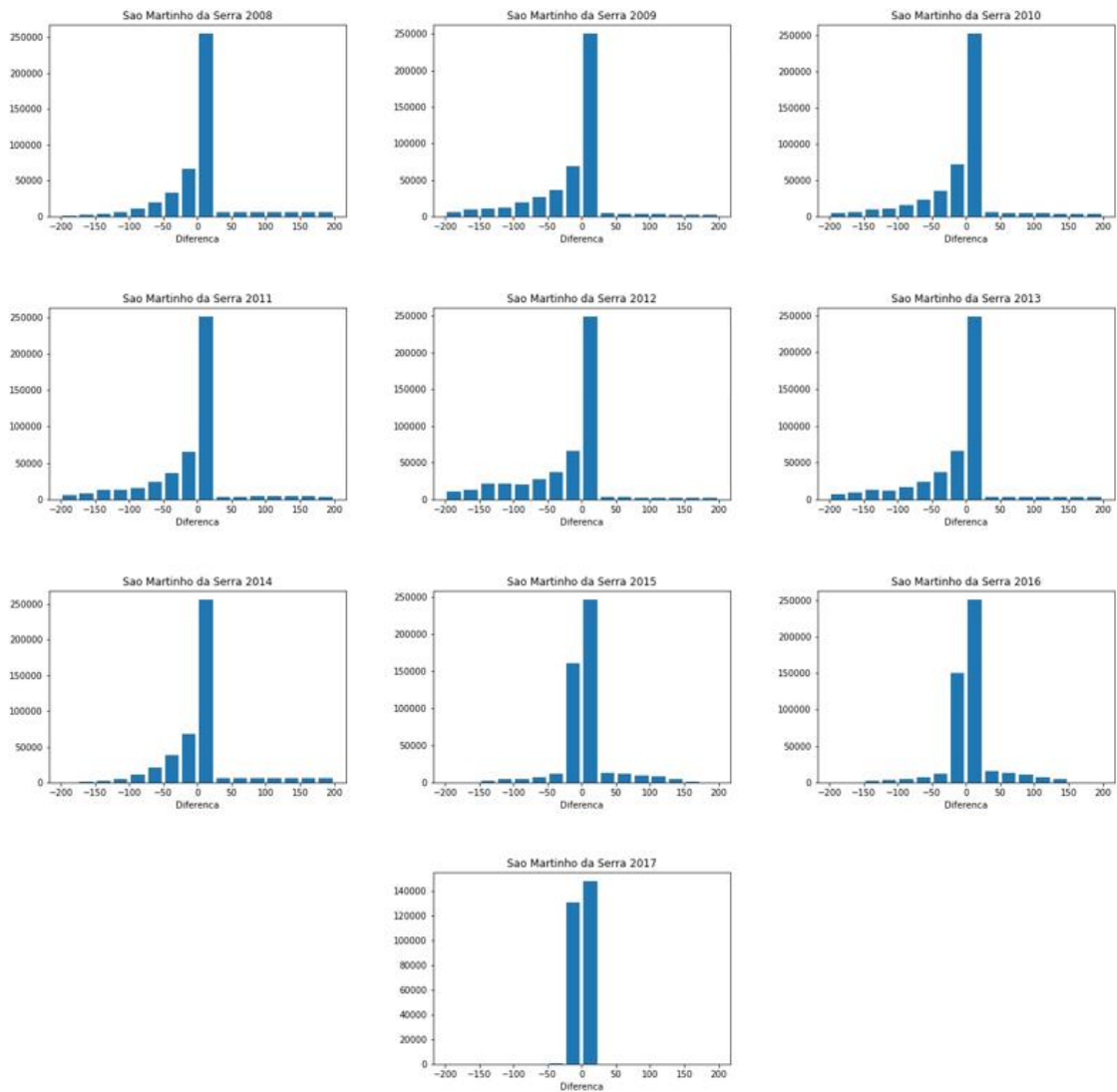


Figura 7. Histogramas da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de São Martinho da Serra, por ano.

A Figura 8 mostra a visão geral da série temporal de dados analisados para a estação de São Martinho da Serra.

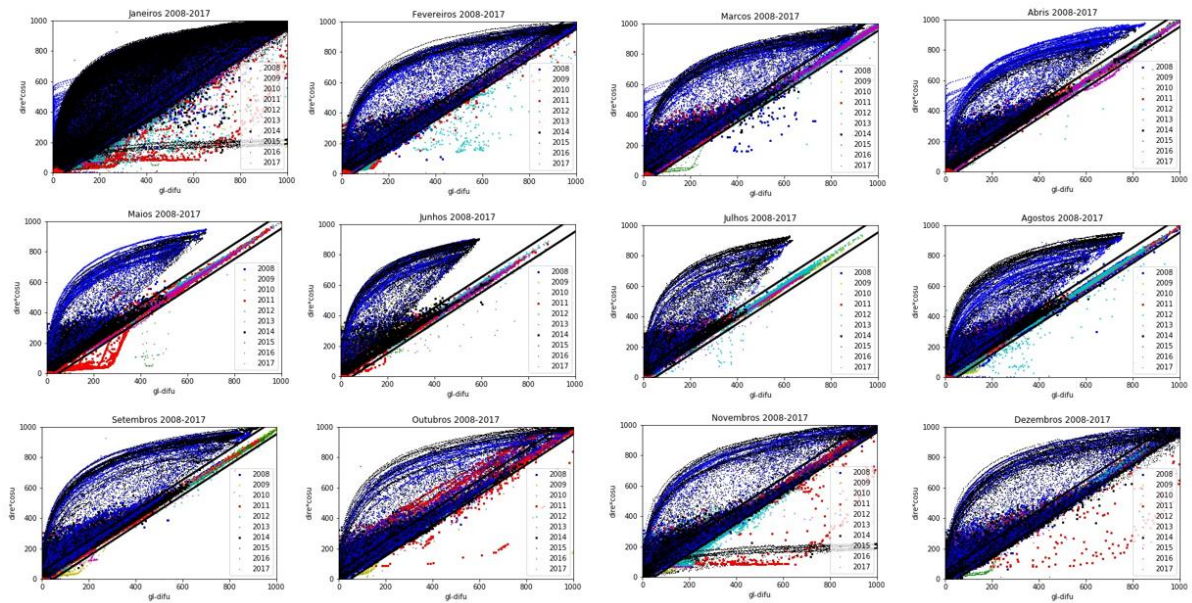


Figura 8. Visão geral da série temporal de dados para a estação de São Martinho da Serra.

4.1.2 Estação de Petrolina

A estação de Petrolina foi a segunda colocada na média geral de dados aprovados pela terceira etapa do processo de qualificação, com 71,32%. Como pode se observar na Tabela 4, a maior média das diferenças aconteceu entre os valores de 0 e 25. Já a menor média das diferenças aconteceu entre os valores de -200 a -176.

Petrolina											
Diferença	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	Média
-200 : -176	0,543%	1,035%	0,449%	0,224%	0,066%	0,021%	0,115%	0,073%	0,066%	0,008%	0,260%
-175 : -151	0,474%	1,264%	0,855%	0,454%	0,268%	0,168%	0,512%	0,094%	0,083%	0,024%	0,420%
-150 : -126	0,538%	1,803%	1,360%	0,961%	0,757%	0,689%	1,006%	0,135%	1,235%	0,396%	0,888%
-125 : -101	1,108%	2,496%	2,252%	1,838%	1,548%	1,635%	1,870%	1,245%	2,243%	2,118%	1,835%
-100 : -76	2,973%	3,397%	3,267%	3,106%	2,087%	2,553%	3,027%	2,374%	2,172%	2,072%	2,703%
-75 : -51	4,538%	4,445%	4,392%	4,397%	3,074%	3,774%	4,305%	3,478%	2,831%	3,307%	3,854%
-50 : -26	5,387%	5,169%	4,893%	4,738%	3,695%	4,698%	4,874%	4,120%	3,833%	4,441%	4,585%
-25 : 0	7,943%	8,237%	7,928%	8,220%	6,608%	7,208%	7,950%	21,366%	56,671%	63,151%	19,528%
0 : 25	51,539%	49,592%	51,045%	51,698%	51,522%	51,073%	50,752%	50,735%	22,247%	17,252%	44,745%
26 : 50	2,736%	2,082%	2,551%	2,456%	2,947%	3,057%	2,499%	5,270%	3,964%	3,064%	3,063%
51 : 75	2,556%	2,437%	2,213%	1,988%	2,785%	2,881%	2,483%	3,237%	1,932%	1,856%	2,437%
76 : 100	2,001%	1,718%	1,845%	1,911%	2,423%	2,187%	1,885%	2,372%	1,211%	1,188%	1,874%
101 : 125	1,820%	1,398%	1,610%	1,827%	2,189%	1,970%	1,734%	0,623%	0,194%	0,112%	1,348%
126 : 150	1,585%	1,238%	1,483%	1,620%	2,304%	1,842%	1,742%	0,035%	0,001%	0,000%	1,185%
151 : 175	1,388%	1,058%	1,307%	1,380%	1,912%	1,814%	1,661%	0,030%	0,000%	0,000%	1,055%
176 : 200	1,208%	1,024%	1,167%	1,315%	1,700%	1,570%	1,324%	0,029%	0,000%	0,000%	0,934%

Tabela 4. Diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Petrolina, por ano.

Os histogramas plotados para a estação são mostrados na Figura 9. Assim como a estação de São Martinho da Serra, pode se observar uma pequena tendência para a parte inferior dos limites de aceitação.

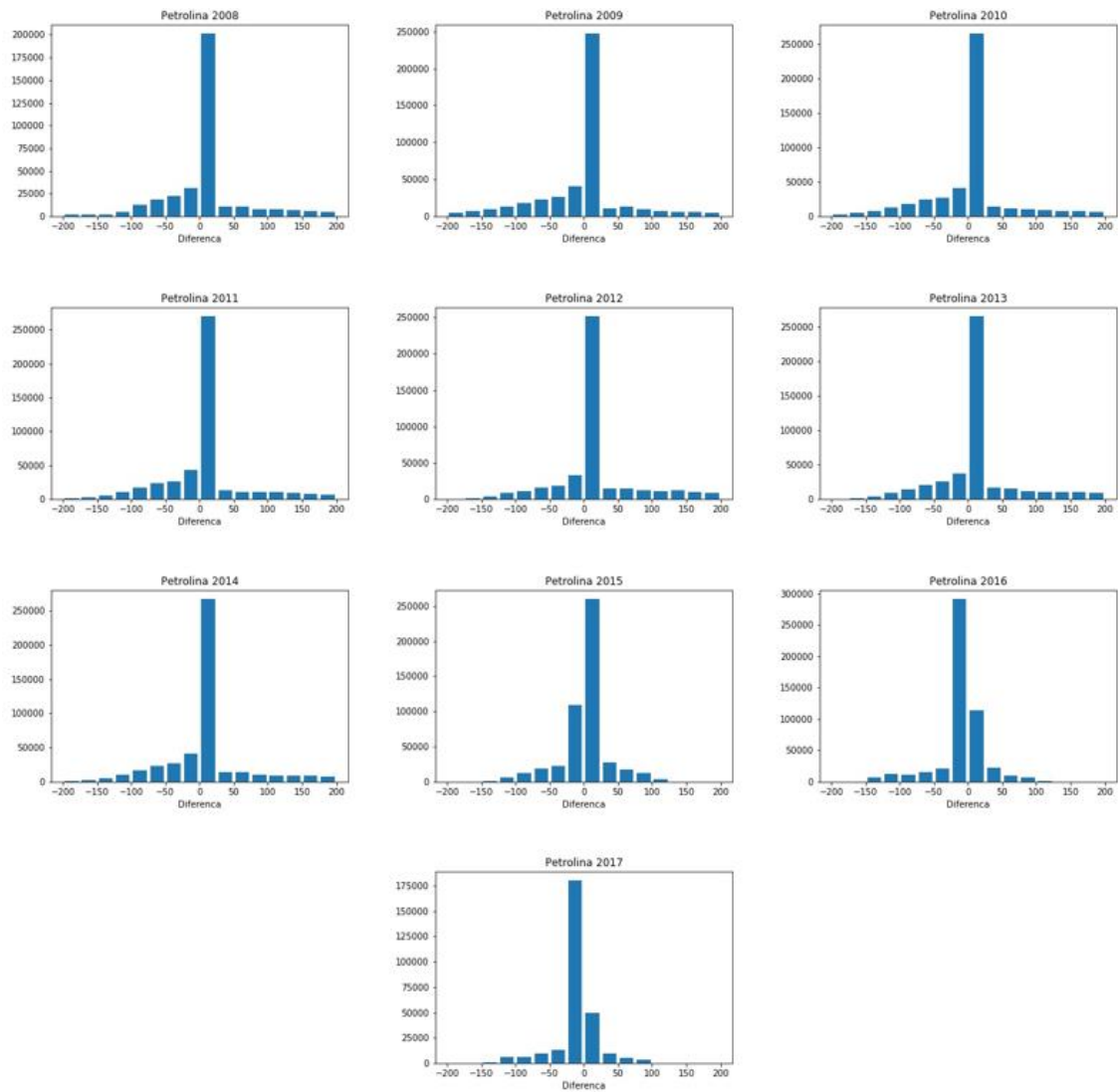


Figura 9. Histogramas da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Petrolina, por ano.

A figura 10 mostra a visão geral da série temporal de dados analisados para a estação de Petrolina.

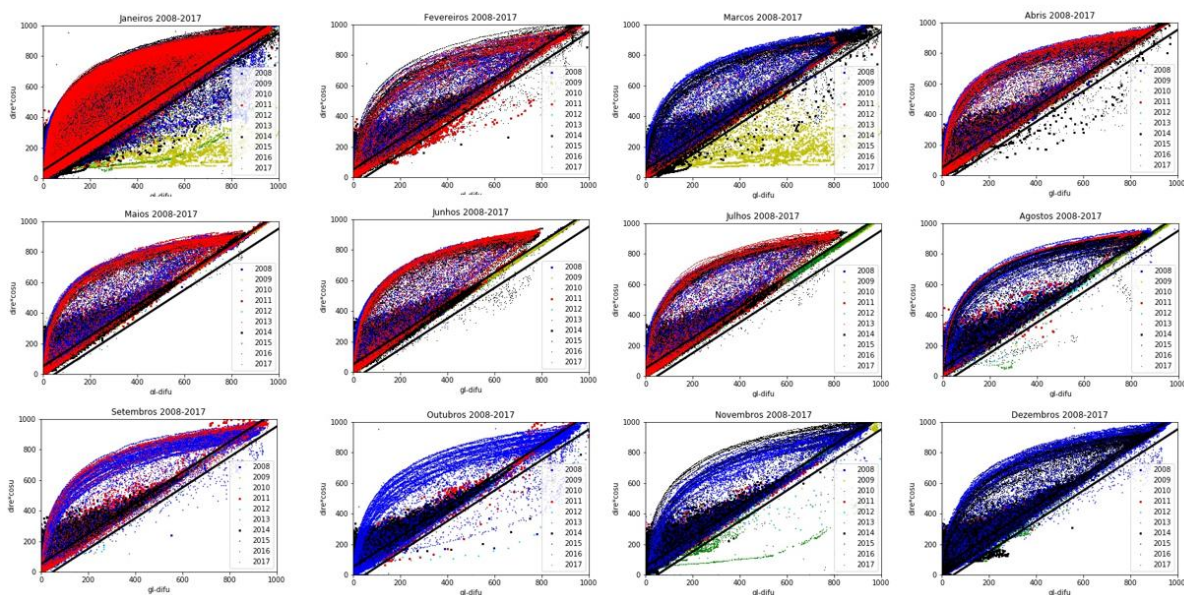


Figura 10. Visão geral da série temporal de dados para a estação de Petrolina.

4.1.3 Estação de Brasília

A estação de Brasília obteve pior média de dados aprovados durante os anos analisados, com apenas 60,23% dos dados validados. Como pode se observar na Tabela 5, a maior média das diferenças ficou entre os valores de 0 e 25, e as menores entre os valores de -200 a -176.

Brasília											
Diferença	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	Média
-200 : -176	0,058%	0,398%	0,018%	0,140%	0,028%	0,101%	0,130%	1,586%	0,807%	0,254%	0,352%
-175 : -151	0,097%	0,284%	0,065%	0,220%	0,057%	0,144%	0,148%	1,507%	0,899%	0,268%	0,369%
-150 : -126	0,498%	0,457%	0,293%	0,601%	0,336%	0,414%	0,378%	1,458%	0,609%	0,270%	0,531%
-125 : -101	1,433%	1,167%	1,035%	1,717%	1,258%	1,278%	1,236%	2,206%	0,915%	0,268%	1,251%
-100 : -76	2,902%	2,132%	2,225%	3,375%	2,960%	2,781%	2,736%	3,213%	1,586%	0,276%	2,419%
-75 : -51	4,592%	3,512%	3,819%	5,607%	4,804%	4,751%	5,001%	4,442%	2,717%	0,303%	3,955%
-50 : -26	6,828%	6,283%	5,725%	8,101%	6,373%	7,257%	6,428%	6,314%	4,085%	1,142%	5,854%
-25 : 0	9,768%	8,007%	10,331%	10,840%	8,676%	9,302%	7,635%	8,693%	6,362%	26,647%	10,626%
0 : 25	49,489%	50,627%	49,349%	49,082%	49,843%	49,753%	44,906%	18,876%	40,504%	21,208%	42,364%
26 : 50	1,503%	1,441%	1,614%	1,280%	1,449%	1,433%	1,670%	1,757%	1,450%	0,299%	1,390%
51 : 75	1,359%	1,483%	1,408%	1,348%	1,613%	1,576%	1,720%	1,556%	1,325%	0,294%	1,368%
76 : 100	1,372%	1,368%	1,416%	1,463%	1,713%	1,606%	1,828%	1,240%	1,406%	0,283%	1,370%
101 : 125	1,425%	1,602%	1,700%	1,544%	1,510%	1,705%	1,894%	1,011%	1,591%	0,286%	1,427%
126 : 150	1,366%	1,813%	1,909%	1,369%	1,474%	1,922%	1,661%	0,922%	1,534%	0,284%	1,425%
151 : 175	1,455%	1,739%	2,171%	1,173%	1,672%	1,525%	1,377%	0,906%	1,496%	0,271%	1,378%
176 : 200	1,472%	1,794%	1,750%	1,089%	1,736%	1,303%	1,303%	0,763%	1,539%	0,279%	1,303%

Tabela 5. Diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Brasília, por ano.

Os histogramas plotados para a estação de Brasília são mostrados na Figura 11. Pode se observar que assim como as outras estações, é possível observar uma tendência para a parte inferior dos limites de aceitação.

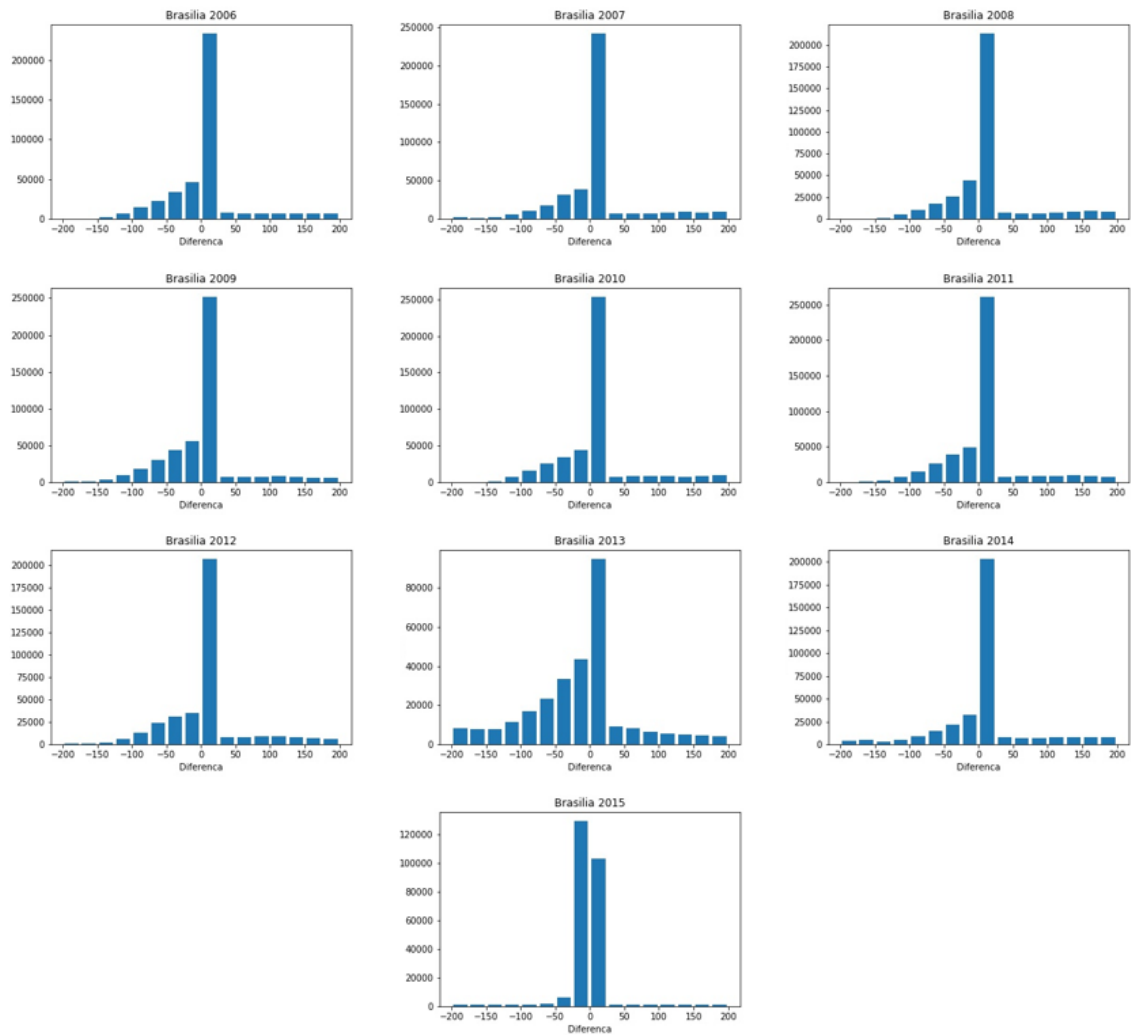


Figura 11. Histogramas da diferença entre valores de Radiação Direta Normal à superfície e valores indiretos da mesma variável obtidos por subtração de dados observados de Radiação Global e Difusa para a estação de Brasília, por ano.

A Figura 12 mostra a visão geral da série temporal de dados analisados para a estação de Brasília.

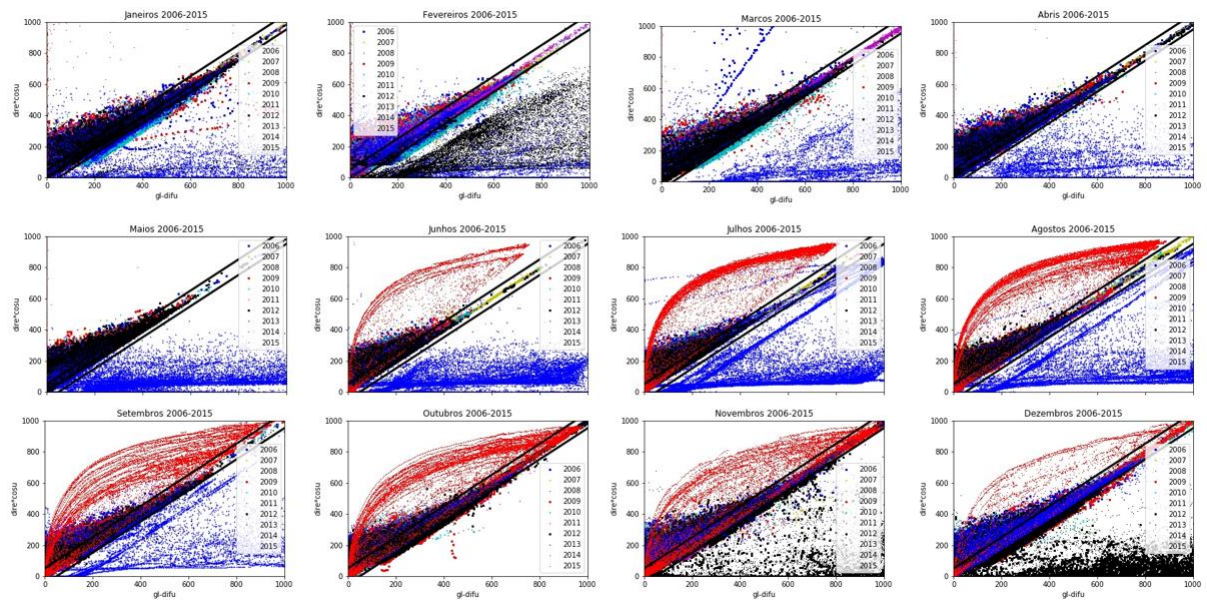


Figura 12. Visão geral da série temporal de dados para a estação de Brasília.

4.2 Análise das inconsistências encontradas

Nesta seção é feita uma análise das inconsistências encontradas nos dados coletados das três estações. A faixa de aceitação indicada pelas linhas pretas inclinadas nos gráficos desta seção refere-se à terceira etapa do processo de qualificação de dados conforme a Equação 1. Os dados observados que aparecem na parte interna entre as duas linhas contínuas são dados validados. Dados fora da região entre as duas linhas são dados suspeitos de pouca confiabilidade e necessitam de maior investigação.

4.2.1 Inconsistência nas medições do pireliômetro

No gráfico da Figura 13, pode se observar que os valores de radiação global e difusa, medidos pelo piranômetro, percorrem todo o eixo das abscissas, enquanto que os valores de radiação direta vezes cosseno do ângulo zenital, no eixo das ordenadas e medidos pelo pireliômetro, não acompanham os valores medidos pelo piranômetro, atingindo um máximo de aproximadamente 300 W/m^2 e fazendo com que os pontos fiquem fora da faixa de aceitação.

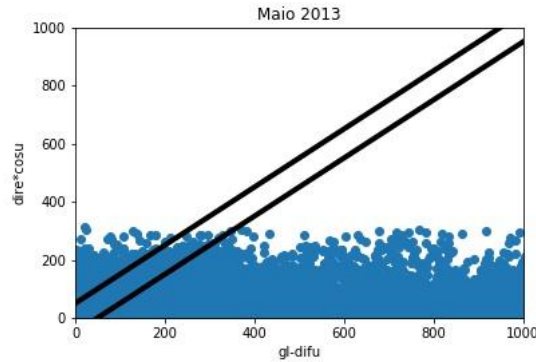


Figura 13. Exemplo de inconsistência nas medições do pireliômetro.

Neste caso, é notável uma inconsistência nos dados coletados pelo pireliômetro. A hipótese mais provável para este fenômeno é o fato de que o sistema de rastreamento do Sol presente no equipamento não estaria funcionando adequadamente, fazendo com que o sensor só medisse uma fração da radiação direta advinda do Sol, pois este não estaria totalmente voltado na direção do astro.

4.2.2 Inconsistência nas medições do piranômetro

No gráfico da Figura 14, é notável que os valores do eixo das ordenadas, medidos pelo pireliômetro, percorrem todo o eixo, enquanto que os valores do eixo das abscissas, medidos pelo piranômetro, são medidos constantemente como zero.

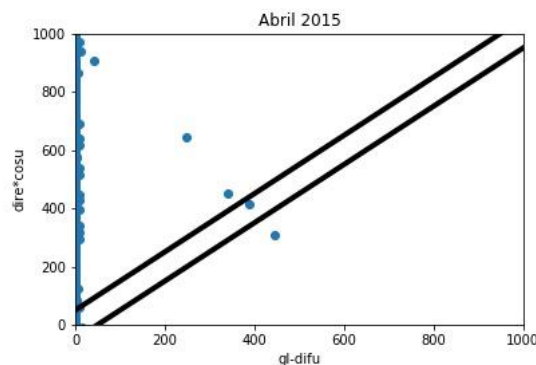


Figura 14. Exemplo de inconsistência nas medições do piranômetro.

Neste caso, temos uma inconsistência nos dados medidos pelo piranômetro. A hipótese mais provável para o fato dos valores de radiação global e difusa estarem constantemente iguais a zero é que o sensor estaria medindo valores iguais, anulando os valores observados. Isto aconteceria numa situação em que o sistema de sombreamento do piranômetro estivesse totalmente desalinhado, fazendo com que o sensor de radiação difusa estivesse medindo os mesmos valores que o sensor de radiação global, pois este

não estaria sendo sombreado em momento algum, portanto captaria toda a componente da radiação emitida pelo Sol.

4.2.3 Inconsistência nas medições de ambos os equipamentos

No gráfico da Figura 15, tanto os valores das abscissas, medidos pelo piranômetro, quanto os valores das ordenadas, medidos pelo pireliômetro, se encontram muito próximos de zero durante todo o período do mês.

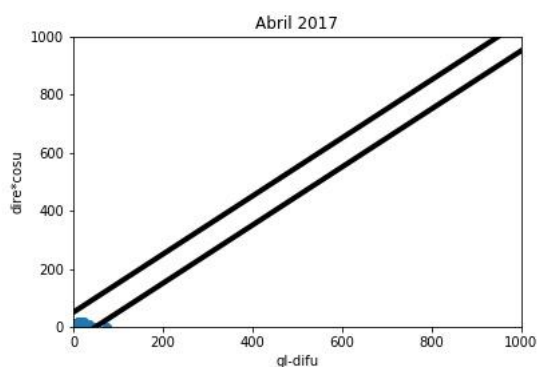


Figura 15. Exemplo de inconsistência nas medições de ambos os equipamentos.

Este é um caso muito peculiar na análise. Como pode se observar, quase que a totalidade dos dados poderiam ser considerados como validados em uma análise superficial. Entretanto, é uma situação impossível fisicamente, pois este gráfico caracterizaria uma situação de céu nublado durante todo o mês na estação de São Martinho da Serra. Então, embora os dados sejam dados como dados validados pelo algoritmo, se analisados a fundo, pode se concluir que são dados de má qualidade e deveriam ser descartados. A hipótese mais provável para este caso é o mal funcionamento de ambos equipamentos, onde o pireliômetro não rastreia o Sol em momento algum, e o piranômetro mede os mesmos valores para radiação global e difusa.

4.2.4 Desvio constante das faixas de aceitação

No gráfico da Figura 16, os valores de radiação direta, global e difusa apresentam considerável consistência; entretanto, a grande maioria desses dados se encontram fora da faixa de aceitação.

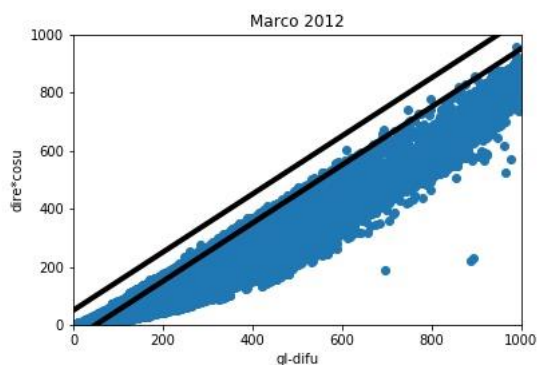


Figura 16. Exemplo de desvio constante das faixas de aceitação.

Como os valores observados apresentam certa consistência e se encontram logo abaixo da faixa de aceitação, acredita-se que os equipamentos de medição estariam com as variáveis matemáticas de calibração desajustadas, fazendo com que os dados coletados sejam considerados de má qualidade, embora os equipamentos estejam funcionando perfeitamente.

4.3 Deficiências encontradas e melhorias propostas

De modo geral, a partir da análise feita, foram encontrados dois principais tipos de deficiências no sistema de coleta de dados no que diz respeito a radiação solarimétrica. Como mostrado, tais erros podem comprometer totalmente a coleta de dados, portanto a identificação e reparo destes se faz necessário de modo a garantir a aquisição de dados de boa qualidade. Ademais, é feita uma análise dos limites utilizados para a terceira etapa do processo de qualificação de dados adotados pela rede SONDA.

4.3.1 Problemas operacionais com os sensores

Problemas operacionais com os sensores podem ocorrer de diversas formas, e podem ser divididos basicamente em problemas de erro na instalação e/ou falhas operacionais e erros de calibração dos equipamentos. Ambos os erros são passíveis de correção, portanto apontá-los e analisá-los se faz necessário de modo a garantir dados solarimétricos de qualidade.

A perda de calibração nos sensores acontece de acordo com a maneira com a qual os equipamentos operam. A calibração dos sensores é feita na etapa de instalação dos equipamentos, através de parâmetros e constantes oferecidas pelo fabricante. O processo de calibração é essencial para a coleta de dados, pois determina o valor da constante que converte a corrente elétrica produzida no sensor por radiação solar, em valores de

Watts/metro². É através dessa constante que todos os dados serão coletados, portanto se houver algum descuido, existe a probabilidade de que todos os dados sejam afetados. Sendo assim, a instalação, calibração e manutenção tem de ser feita com cautela e por pessoal que tenha experiência com o equipamento e o procedimento, assegurando assim a qualidade dos dados coletados.

Com relação aos erros de instalação e falhas operacionais, cada instrumento de medição tem suas peculiaridades. No caso do pireliômetro, se o sensor termoeletrico não estiver apontando diretamente para o Sol - o que acontece quando o sistema de rastreamento não funciona perfeitamente - os dados de radiação direta são prejudicados. Neste tipo de caso, se o rastreador estiver moderadamente desviado, o equipamento não medirá a radiação direta em sua totalidade, pois o sensor só estará captando parte da radiação solar, causando uma pequena diminuição dos valores medidos. Em casos extremos, se o sensor estiver totalmente desalinhado, o equipamento terá como saídas dados de radiação difusa como sendo radiação direta, pois o sensor não estaria captando nenhuma parte da radiação direta advinda do Sol. Além disso, podemos ter erros de calibração do equipamento no que diz respeito ao software instalado e suas constantes matemáticas utilizadas.

Já no caso dos piranômetros, podemos ter falhas no que diz respeito a sua instalação. Se o equipamento não estiver posicionado na horizontal, por exemplo, os dados de irradiação global serão afetados. Este equipamento também pode possuir acoplado um sistema de sombreamento para medição da radiação difusa. Neste caso, se o sistema de sombreamento não estiver bem ajustado com relação a trajetória do Sol no céu, o equipamento pode medir erroneamente os valores de radiação difusa. No pior dos casos, no total desalinhamento deste, não causando sombreamento nenhum do sensor, teríamos medições de radiação global sendo coletadas como dados de radiação difusa.

Portanto, de modo a minimizar os erros de coleta de dados no caso de erros de instalação e falhas operacionais, é necessário principalmente a correta instalação dos equipamentos por pessoal qualificado. Além disso, a investigação e acompanhamento constante dos equipamentos a fim de corrigir erros prontamente é outro procedimento que deve ser implementado.

4.3.2 Falta de manutenção nos equipamentos

O segundo principal erro encontrado foi a falta de manutenção ou manutenção precária dos equipamentos das estações solarimétricas. Os piranômetros e pireliômetros são equipamentos que precisam de manutenção constante, pois apresentam dispositivos

sensíveis e que se danificados ou modificados prejudicam totalmente a coleta de dados. Desse modo, a manutenção constante dos equipamentos se faz necessária, de forma a manter um padrão de qualidade nos dados medidos. Em alguns casos, por exemplo, dados são coletados precariamente por meses até que ocorra a manutenção dos equipamentos e os mesmos voltem a funcionar adequadamente. Assim sendo, uma revisão do padrão de manutenção das estações da rede SONDA precisa ser feita de modo a sanar esses problemas recorrentes.

4.3.3 Análise dos limites utilizados nos critérios de qualificação de dados

Como mostrado em seções anteriores, os limites de aceitação dos dados solarimétricos mostrados neste trabalho foram adotados com base na Equação 1, baseada em critérios das organizações BSRN e webmet.com. Além disso, foi mostrado como, na grande maioria dos anos, os dados sempre tendem para o limite inferior do limite de aceitação. A partir dessa ideia, foram elaboradas tabelas mostrando a soma de porcentagens entre os limites de +50 e -50 para cada ano de cada estação, segundo os limites adotados, e foi criado um “novo limite”, mantendo a variação de 100 do critério atual, entretanto utilizando um limite de +25 e -75. As Tabelas 6, 7 e 8, mostram esses resultados.

	São Martinho da Serra										
	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	Media
Soma das porcentagens entre +50 e -50	68,736%	68,188%	69,258%	67,802%	67,261%	68,634%	70,205%	88,277%	87,913%	97,152%	75,343%
Soma das porcentagens entre +25 e -75	71,030%	72,326%	72,417%	71,544%	71,787%	72,272%	72,897%	87,122%	86,153%	97,170%	77,472%

Tabela 6. Soma das porcentagens e média geral para cada limite, por ano, para a estação de São Martinho da Serra.

	Brasília										
	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	Media
Soma das porcentagens entre +50 e -50	67,588%	66,359%	67,019%	69,303%	66,341%	67,745%	60,639%	35,640%	52,401%	49,296%	60,233%
Soma das porcentagens entre +25 e -75	70,677%	68,429%	69,224%	73,630%	69,696%	71,062%	63,970%	38,325%	53,668%	49,300%	62,798%

Tabela 7. Soma das porcentagens e média geral para cada limite, por ano, para a estação de Brasília.

	Petrolina										
	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	Media
Soma das porcentagens entre +50 e -50	67,605%	65,079%	66,416%	67,112%	64,773%	66,036%	66,075%	81,491%	86,715%	87,907%	71,921%
Soma das porcentagens entre +25 e -75	69,408%	67,443%	68,257%	69,053%	64,899%	66,753%	67,882%	79,698%	85,582%	88,150%	72,712%

Tabela 8. Soma das porcentagens e média geral para cada limite, por ano, para a estação de Petrolina.

Fica claro que em grande parte dos anos a soma das porcentagens entre +25 e -75 - ou seja, se tomada a parte inferior dos limites de aceitação - é maior. Uma hipótese para esse tipo de resultado, é o fato de que os limites de aceitação adotados pela rede SONDA possam ser falhos. Mais especificamente, estes limites foram elaborados para países de clima temperado, e possivelmente não funcionariam adequadamente para outras localidades com condições climáticas e posições geográficas diferentes. Portanto, talvez

a revisão dos limites adotados seja necessária, de modo a adequar esses limites para a adoção de um método de qualificação mais efetivo para o território brasileiro, garantindo a disponibilização de dados de melhor qualidade para o setor acadêmico e empresarial.

5. CONSIDERAÇÕES FINAIS

O processo de qualificação de dados da rede SONDA tem um papel importante na disponibilização de dados para a comunidade do setor acadêmico e do setor de energia. Este trabalho teve como intuito fazer um estudo e análise geral do processo de qualificação dos dados, mais precisamente a terceira etapa do processo de qualificação de dados da rede SONDA, operada pelo Instituto Nacional de Pesquisas Espaciais. A partir da análise elaborada, foi possível propor melhorias e apontar pontos críticos ao processo de qualificação, de modo a aumentar a confiabilidade dos dados disponibilizados. Os principais pontos críticos encontrados foram problemas nos rastreadores e sensores e falta de manutenção dos equipamentos de coleta de dados. Dessa forma, foram propostas medidas para melhoria da coleta de dados, como por exemplo a maior frequência no acompanhamento e manutenção das estações solarimétricas. Um possível ponto a ser analisado em próximos trabalhos é o ajuste dos limites utilizados nos critérios de qualificação de dados para condições climáticas características do Brasil, visto que condições climáticas poderiam ter impacto na coleta de dados, e os critérios de aceitação utilizados pela rede SONDA são baseados em padrões internacionais que foram desenvolvidos em regiões de zona temperada.

REFERÊNCIAS BIBLIOGRÁFICAS

Agência Nacional de Energia Elétrica. **Relatórios e retrospectivas**. Disponível em: <<http://www.aneel.gov.br/>>. Acesso em: 05 jun. 2018.

BENTLEY, R. W. **Global oil & gas depletion: an overview**, Energy Policy, Volume 30, Issue 3, 2002, Pages 189-205, ISSN 0301-4215. Disponível em: <[https://doi.org/10.1016/S0301-4215\(01\)00144-6](https://doi.org/10.1016/S0301-4215(01)00144-6)>. Acesso em: 05 jun. 2018.

GELLER, H. S., **Revolução energética: políticas para um futuro sustentável**. Tese submetida ao Programa Interunidades de Pós-Graduação em Energia da Universidade de São Paulo. São Paulo, Julho de 2012. Disponível em: <<http://www.iee.usp.br/producao/2002/Teses/Geller.pdf>>. Acesso em: 6 jun. 2018.

GOLDEMBERG, J., VILLANUEVA, L. D. **Energia, meio ambiente & desenvolvimento**. 2.ed. São Paulo: EDUSP, 2003. 226p.

Instituto Nacional de Pesquisas Espaciais. **Sistema de Organização Nacional de Dados Ambientais**. Disponível em: <<http://sonda.ccst.inpe.br/index.html>>. Acesso em: 10 jun. 2018.

IQBAL, M. **An introduction to solar radiation**. Elsevier, 2012.

KUMAR, A., KUMAR, K., KAUSHIK, N., SHARMA, S., MISHRA, S. **Renewable energy in India: Current status and future potentials**, Renewable and Sustainable Energy Reviews, Volume 14, Issue 8, 2010, Pages 2434-2442, ISSN 1364-0321. Disponível em: <<https://doi.org/10.1016/j.rser.2010.04.003>>. Acesso em: 5 jun. 2018.

MARTINS, F., RUTHER, R., PEREIRA, E. B., ABREU, S. **Solar energy scenarios in Brazil. Part two: Photovoltaics applications**. Energy Policy, v. 36, n. 8, p. 2865-2877, 2008.

PEREIRA, E. B., MARTINS, F., THOMAZ, J. C., NOGUEIRA, J. **Treinamento teórico-prático em avaliação de potencial de recurso energético solar**. Material didático elaborado para curso oferecido no Centro de Tecnologias do Gás e Energias Renováveis (CTGÁS ER), 2014.

APÊNDICE A – CÓDIGO DE ELABORAÇÃO DOS GRÁFICOS

```
#PETROLINA
import pandas as pd
import numpy as np
import math as math
import matplotlib.pyplot as plt
#####
#####
#UPLOAD ARQUIVOS
b2008=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2008.txt', delimiter = ',')
b2009=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2009.txt', delimiter = ',')
b2010=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2010.txt', delimiter = ',')
b2011=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2011.txt', delimiter = ',')
b2012=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2012.txt', delimiter = ',')
b2013=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2013.txt', delimiter = ',')
b2014=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2014.txt', delimiter = ',')
b2015=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2015.txt', delimiter = ',')
b2016=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2016.txt', delimiter = ',')
b2017=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\PTR\PTR_2017.txt', delimiter = ',')
b2008 = b2008.dropna(axis=0)
b2009 = b2009.dropna(axis=0)
b2010 = b2010.dropna(axis=0)
b2011 = b2011.dropna(axis=0)
b2012 = b2012.dropna(axis=0)
b2013 = b2013.dropna(axis=0)
b2014 = b2014.dropna(axis=0)
b2015 = b2015.dropna(axis=0)
b2016 = b2016.dropna(axis=0)
b2017 = b2017.dropna(axis=0)
b2008=b2008.values
b2009=b2009.values
b2010=b2010.values
b2011=b2011.values
b2012=b2012.values
b2013=b2013.values
b2014=b2014.values
b2015=b2015.values
b2016=b2016.values
b2017=b2017.values
#####
#####
#CRIACAO DAS MATRIZES
anos = np.concatenate((b2008,b2009,b2010,b2011,b2012,b2013,b2014,b2015,b2016,b2017), axis = 0)
size=len(anos)
countj=0
for i in range (1,32):
    for j in range (0,size):
        if anos[j,2] == i:
            countj=countj+1
janeiros = np.zeros((countj,7))
```

```

k=0
for i in range (0,size):
    if anos[i,2] <= 31:
        janeiros[k,:]= anos[i,:]
        k=k+1
#FEVEREIRO
countf=0
for i in range (32,60):
    for j in range (0,size):
        if anos[j,2] == i:
            countf=countf+1
fevereiros = np.zeros((countf,7))
k=0
for i in range (0,size):
    if anos[i,2] >=32 and anos[i,2]<60:
        fevereiros[k,:]= anos[i,:]
        k=k+1
#MARCOS
countm=0
for i in range (60,91):
    for j in range (0,size):
        if anos[j,2] == i:
            countm=countm+1
marcos = np.zeros((countm,7))
k=0
for i in range (0,size):
    if anos[i,2] >=60 and anos[i,2]<91:
        marcos[k,:]= anos[i,:]
        k=k+1
#ABRIL
counta=0
for i in range (91,121):
    for j in range (0,size):
        if anos[j,2] == i:
            counta=counta+1
abril = np.zeros((counta,7))
k=0
for i in range (0,size):
    if anos[i,2] >=91 and anos[i,2]<121:
        abril[k,:]= anos[i,:]
        k=k+1
#MAIO
countmai=0
for i in range (121,152):
    for j in range (0,size):
        if anos[j,2] == i:
            countmai=countmai+1
maios = np.zeros((countmai,7))
k=0
for i in range (0,size):
    if anos[i,2] >=121 and anos[i,2]<152:
        maios[k,:]= anos[i,:]
        k=k+1
#JUNHO
countjun=0
for i in range (152,182):
    for j in range (0,size):
        if anos[j,2] == i:
            countjun=countjun+1
junhos = np.zeros((countjun,7))

```

```

k=0
for i in range (0,size):
    if anos[i,2] >=152 and anos[i,2]<182:
        junhos[k,:]= anos[i,:]
        k=k+1
#JULHOS
countjl=0
for i in range (182,213):
    for j in range (0,size):
        if anos[j,2] == i:
            countjl=countjl+1
julhos = np.zeros((countjl,7))
k=0
for i in range (0,size):
    if anos[i,2] >=182 and anos[i,2]<213:
        julhos[k,:]= anos[i,:]
        k=k+1
#AGOSTOS
countago=0
for i in range (213,244):
    for j in range (0,size):
        if anos[j,2] == i:
            countago=countago+1
agostos = np.zeros((countago,7))
k=0
for i in range (0,size):
    if anos[i,2] >=213 and anos[i,2]<244:
        agostos[k,:]= anos[i,:]
        k=k+1
#SETEMBROS
countset=0
for i in range (244,274):
    for j in range (0,size):
        if anos[j,2] == i:
            countset=countset+1
setembros = np.zeros((countset,7))
k=0
for i in range (0,size):
    if anos[i,2] >=244 and anos[i,2]<274:
        setembros[k,:]= anos[i,:]
        k=k+1
#OUTUBROS
countout=0
for i in range (274,305):
    for j in range (0,size):
        if anos[j,2] == i:
            countout=countout+1
outubros = np.zeros((countout,7))
k=0
for i in range (0,size):
    if anos[i,2] >=274 and anos[i,2]<305:
        outubros[k,:]= anos[i,:]
        k=k+1
#NOVEMBROS
countnov=0
for i in range (305,335):
    for j in range (0,size):
        if anos[j,2] == i:
            countnov=countnov+1
novembros = np.zeros((countnov,7))

```

```

k=0
for i in range (0,size):
    if anos[i,2] >=305 and anos[i,2]<335:
        novembros[k,:]= anos[i,:]
        k=k+1
#DEZEMBROS
countdez=0
for i in range (335,366):
    for j in range (0,size):
        if anos[j,2] == i:
            countdez=countdez+1
dezembros = np.zeros((countdez,7))
k=0
for i in range (0,size):
    if anos[i,2] >=335 and anos[i,2]<366:
        dezembros[k,:]= anos[i,:]
        k=k+1
#####
#####
#CALCULOS
lat=-29.442777777778
lon=-53.823055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
mat=np.concatenate((janeiros,fevereiros,marcos,abris,maios,junhos,julhos,agostos,setembros,outubros,no
vembros,dezembros), axis=0)
dia=mat[:,2]
minuto=mat[:,3]
gl=mat[:,4]
dire=mat[:,6]
difu=mat[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
final=np.append(mat,cosu[:,None], axis=1)
#####
#####
#JANEIROS
sizej=len(final)

countj=0
for i in range (2008, 2009):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2008 = np.zeros((countj,8))
k=0

```

```

for i in range (0,sizej):
    if final[i,1] == 2008:
        jan_2008[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2009, 2010):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2009 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2009:
        jan_2009[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2010, 2011):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2010 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2010:
        jan_2010[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2011, 2012):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2011 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2011:
        jan_2011[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2012, 2013):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2012 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2012:
        jan_2012[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2013, 2014):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2013 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2013:
        jan_2013[k,:]= final[i,:]
        k=k+1
countj=0

```

```

for i in range (2014, 2015):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2014 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2014:
        jan_2014[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2015, 2016):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2015 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2015:
        jan_2015[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2016, 2017):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2016 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2016:
        jan_2016[k,:]= final[i,:]
        k=k+1
countj=0
for i in range (2017, 2018):
    for j in range (0,sizej):
        if final[j,1] == i:
            countj=countj+1
jan_2017 = np.zeros((countj,8))
k=0
for i in range (0,sizej):
    if final[i,1] == 2017:
        jan_2017[k,:]= final[i,:]
        k=k+1

#####
#####
#FEVEREIROS
sizen=len(final)
countf2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2008=countf2008+1

fev_2008 = np.zeros((countf2008,8))
k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2008[k,:]= final[j,:]

```



```

    k=k+1

countf2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2009=countf2009+1

fev_2009 = np.zeros((countf2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2009[k,:]= final[j,:]
            k=k+1

countf2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2010=countf2010+1

fev_2010 = np.zeros((countf2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2010[k,:]= final[j,:]
            k=k+1

countf2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2011=countf2011+1

fev_2011 = np.zeros((countf2011,8))
k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2011[k,:]= final[j,:]
            k=k+1

countf2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2012=countf2012+1

fev_2012 = np.zeros((countf2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2012[k,:]= final[j,:]
            k=k+1

```

```

countf2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2013=countf2013+1

fev_2013 = np.zeros((countf2013,8))

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2013[k,:]= final[j,:]
            k=k+1

countf2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2014=countf2014+1

fev_2014 = np.zeros((countf2014,8))
k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2014[k,:]= final[j,:]
            k=k+1

countf2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2015=countf2015+1

fev_2015 = np.zeros((countf2015,8))

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2015[k,:]= final[j,:]
            k=k+1

countf2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2016=countf2016+1

fev_2016 = np.zeros((countf2016,8))

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2016[k,:]= final[j,:]
            k=k+1

```

```

countf2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            countf2017=countf2017+1

fev_2017 = np.zeros((countf2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=32 and final[j,2]<60:
            fev_2017[k,:]= final[j,:]
            k=k+1

#####
#####
#MARCOS
countm2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2008=countm2008+1

mar_2008 = np.zeros((countm2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2008[k,:]= final[j,:]
            k=k+1

countm2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2009=countm2009+1

mar_2009 = np.zeros((countm2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2009[k,:]= final[j,:]
            k=k+1

countm2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2010=countm2010+1

mar_2010 = np.zeros((countm2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:

```

```

    mar_2010[k,:]= final[j,:]
    k=k+1

countm2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2011=countm2011+1

mar_2011 = np.zeros((countm2011,8))

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2011[k,:]= final[j,:]
            k=k+1

countm2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2012=countm2012+1

mar_2012 = np.zeros((countm2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2012[k,:]= final[j,:]
            k=k+1

countm2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2013=countm2013+1

mar_2013 = np.zeros((countm2013,8))

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2013[k,:]= final[j,:]
            k=k+1

countm2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2014=countm2014+1

mar_2014 = np.zeros((countm2014,8))

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:

```

```

    mar_2014[k,:]= final[j,:]
    k=k+1

countm2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2015=countm2015+1

mar_2015 = np.zeros((countm2015,8))

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2015[k,:]= final[j,:]
            k=k+1

countm2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2016=countm2016+1

mar_2016 = np.zeros((countm2016,8))

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2016[k,:]= final[j,:]
            k=k+1

countm2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            countm2017=countm2017+1

mar_2017 = np.zeros((countm2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=60 and final[j,2]<91:
            mar_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#ABRIS
counta2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2008=counta2008+1

abr_2008 = np.zeros((counta2008,8))

k=0

```

```

for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2008[k,:]= final[j,:]
            k=k+1

counta2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2009=counta2009+1

abr_2009 = np.zeros((counta2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2009[k,:]= final[j,:]
            k=k+1

counta2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2010=counta2010+1

abr_2010 = np.zeros((counta2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2010[k,:]= final[j,:]
            k=k+1

counta2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2011=counta2011+1

abr_2011 = np.zeros((counta2011,8))

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2011[k,:]= final[j,:]
            k=k+1

counta2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2012=counta2012+1

```

```

abr_2012 = np.zeros((counta2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2012[k,:]= final[j,:]
            k=k+1

counta2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2013=counta2013+1

abr_2013 = np.zeros((counta2013,8))

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2013[k,:]= final[j,:]
            k=k+1

counta2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2014=counta2014+1

abr_2014 = np.zeros((counta2014,8))

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2014[k,:]= final[j,:]
            k=k+1

counta2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2015=counta2015+1

abr_2015 = np.zeros((counta2015,8))

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2015[k,:]= final[j,:]
            k=k+1

counta2016=0
for i in range (2016, 2017):

```

```

for j in range (0,sizen):
    if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
        counta2016=counta2016+1

abr_2016 = np.zeros((counta2016,8))

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2016[k,:]= final[j,:]
            k=k+1

counta2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            counta2017=counta2017+1

abr_2017 = np.zeros((counta2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=91 and final[j,2]<121:
            abr_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#MAIOS
countmai2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2008=countmai2008+1

mai_2008 = np.zeros((countmai2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2008[k,:]= final[j,:]
            k=k+1

countmai2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2009=countmai2009+1

mai_2009 = np.zeros((countmai2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):

```



```

    if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
        mai_2009[k,:]= final[j,:]
        k=k+1

countmai2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2010=countmai2010+1

mai_2010 = np.zeros((countmai2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2010[k,:]= final[j,:]
            k=k+1
countmai2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2011=countmai2011+1

mai_2011 = np.zeros((countmai2011,8))

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2011[k,:]= final[j,:]
            k=k+1

countmai2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2012=countmai2012+1

mai_2012 = np.zeros((countmai2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2012[k,:]= final[j,:]
            k=k+1

countmai2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2013=countmai2013+1

mai_2013 = np.zeros((countmai2013,8))

```

```

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2013[k,:]= final[j,:]
            k=k+1
countmai2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2014=countmai2014+1

```

```

mai_2014 = np.zeros((countmai2014,8))

```

```

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2014[k,:]= final[j,:]
            k=k+1

```

```

countmai2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2015=countmai2015+1

```

```

mai_2015 = np.zeros((countmai2015,8))

```

```

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2015[k,:]= final[j,:]
            k=k+1

```

```

countmai2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2016=countmai2016+1

```

```

mai_2016 = np.zeros((countmai2016,8))

```

```

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2016[k,:]= final[j,:]
            k=k+1

```

```

countmai2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            countmai2017=countmai2017+1

```

```

mai_2017 = np.zeros((countmai2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=121 and final[j,2]<152:
            mai_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#JUNHOS
countjun2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2008=countjun2008+1

jun_2008 = np.zeros((countjun2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2008[k,:]= final[j,:]
            k=k+1

countjun2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2009=countjun2009+1

jun_2009 = np.zeros((countjun2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2009[k,:]= final[j,:]
            k=k+1

countjun2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2010=countjun2010+1

jun_2010 = np.zeros((countjun2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2010[k,:]= final[j,:]
            k=k+1
countjun2011=0

```

```

for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2011=countjun2011+1

```

```

jun_2011 = np.zeros((countjun2011,8))

```

```

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2011[k,:]= final[j,:]
            k=k+1

```

```

countjun2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2012=countjun2012+1

```

```

jun_2012 = np.zeros((countjun2012,8))

```

```

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2012[k,:]= final[j,:]
            k=k+1

```

```

countjun2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2013=countjun2013+1

```

```

jun_2013 = np.zeros((countjun2013,8))

```

```

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2013[k,:]= final[j,:]
            k=k+1

```

```

countjun2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2014=countjun2014+1

```

```

jun_2014 = np.zeros((countjun2014,8))

```

```

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2014[k,:]= final[j,:]

```

```

k=k+1

countjun2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2015=countjun2015+1

jun_2015 = np.zeros((countjun2015,8))

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2015[k,:]= final[j,:]
            k=k+1

countjun2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2016=countjun2016+1

jun_2016 = np.zeros((countjun2016,8))

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2016[k,:]= final[j,:]
            k=k+1
countjun2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            countjun2017=countjun2017+1

jun_2017 = np.zeros((countjun2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=152 and final[j,2]<182:
            jun_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#JULHOS
countjul2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2008=countjul2008+1

jul_2008 = np.zeros((countjul2008,8))

k=0

```

```

for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2008[k,:]= final[j,:]
            k=k+1

```

```

countjul2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2009=countjul2009+1

```

```

jul_2009 = np.zeros((countjul2009,8))

```

```

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2009[k,:]= final[j,:]
            k=k+1

```

```

countjul2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2010=countjul2010+1

```

```

jul_2010 = np.zeros((countjul2010,8))

```

```

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2010[k,:]= final[j,:]
            k=k+1

```

```

countjul2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2011=countjul2011+1

```

```

jul_2011 = np.zeros((countjul2011,8))

```

```

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2011[k,:]= final[j,:]
            k=k+1

```

```

countjul2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2012=countjul2012+1

```

```

jul_2012 = np.zeros((countjul2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2012[k,:]= final[j,:]
            k=k+1

countjul2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2013=countjul2013+1

jul_2013 = np.zeros((countjul2013,8))

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2013[k,:]= final[j,:]
            k=k+1
countjul2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2014=countjul2014+1

jul_2014 = np.zeros((countjul2014,8))

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2014[k,:]= final[j,:]
            k=k+1

countjul2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2015=countjul2015+1

jul_2015 = np.zeros((countjul2015,8))

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2015[k,:]= final[j,:]
            k=k+1

countjul2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:

```

```

countjul2016=countjul2016+1

jul_2016 = np.zeros((countjul2016,8))

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2016[k,:]= final[j,:]
            k=k+1

countjul2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            countjul2017=countjul2017+1

jul_2017 = np.zeros((countjul2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=182 and final[j,2]<213:
            jul_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#AGOSTOS
countag2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2008=countag2008+1

ago_2008 = np.zeros((countag2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2008[k,:]= final[j,:]
            k=k+1

countag2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2009=countag2009+1

ago_2009 = np.zeros((countag2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2009[k,:]= final[j,:]

```



```

k=k+1

countag2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2010=countag2010+1

ago_2010 = np.zeros((countag2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2010[k,:]= final[j,:]
            k=k+1
countag2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2011=countag2011+1

ago_2011 = np.zeros((countag2011,8))

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2011[k,:]= final[j,:]
            k=k+1

countag2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2012=countag2012+1

ago_2012 = np.zeros((countag2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2012[k,:]= final[j,:]
            k=k+1

countag2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2013=countag2013+1

ago_2013 = np.zeros((countag2013,8))

k=0
for i in range (2013, 2014):

```

```

for j in range (0,sizen):
    if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
        ago_2013[k,:]= final[j,:]
        k=k+1
countag2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2014=countag2014+1

```

```
ago_2014 = np.zeros((countag2014,8))
```

```

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2014[k,:]= final[j,:]
            k=k+1

```

```

countag2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2015=countag2015+1

```

```
ago_2015 = np.zeros((countag2015,8))
```

```

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2015[k,:]= final[j,:]
            k=k+1

```

```

countag2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2016=countag2016+1

```

```
ago_2016 = np.zeros((countag2016,8))
```

```

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2016[k,:]= final[j,:]
            k=k+1

```

```

countag2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            countag2017=countag2017+1

```

```
ago_2017 = np.zeros((countag2017,8))
```

```

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=213 and final[j,2]<244:
            ago_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#SETEMBROS
counts2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            counts2008=counts2008+1

set_2008 = np.zeros((counts2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            set_2008[k,:]= final[j,:]
            k=k+1

counts2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            counts2009=counts2009+1

set_2009 = np.zeros((counts2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            set_2009[k,:]= final[j,:]
            k=k+1

counts2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            counts2010=counts2010+1

set_2010 = np.zeros((counts2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            set_2010[k,:]= final[j,:]
            k=k+1
counts2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):

```

```
if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
    counts2011=counts2011+1
```

```
set_2011 = np.zeros((counts2011,8))
```

```
k=0  
for i in range (2011, 2012):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            set_2011[k,:]= final[j,:]  
            k=k+1
```

```
counts2012=0  
for i in range (2012, 2013):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            counts2012=counts2012+1
```

```
set_2012 = np.zeros((counts2012,8))
```

```
k=0  
for i in range (2012, 2013):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            set_2012[k,:]= final[j,:]  
            k=k+1
```

```
counts2013=0  
for i in range (2013, 2014):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            counts2013=counts2013+1
```

```
set_2013 = np.zeros((counts2013,8))
```

```
k=0  
for i in range (2013, 2014):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            set_2013[k,:]= final[j,:]  
            k=k+1
```

```
counts2014=0  
for i in range (2014, 2015):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            counts2014=counts2014+1
```

```
set_2014 = np.zeros((counts2014,8))
```

```
k=0  
for i in range (2014, 2015):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:  
            set_2014[k,:]= final[j,:]  
            k=k+1
```

```

counts2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            counts2015=counts2015+1

```

```

set_2015 = np.zeros((counts2015,8))

```

```

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            set_2015[k,:]= final[j,:]
            k=k+1

```

```

counts2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            counts2016=counts2016+1

```

```

set_2016 = np.zeros((counts2016,8))

```

```

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            set_2016[k,:]= final[j,:]
            k=k+1

```

```

counts2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            counts2017=counts2017+1

```

```

set_2017 = np.zeros((counts2017,8))

```

```

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=244 and final[j,2]<274:
            set_2017[k,:]= final[j,:]
            k=k+1

```

```

#####
#####

```

```

#OUTUBROS

```

```

countout2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2008=countout2008+1

```

```

out_2008 = np.zeros((countout2008,8))

```

```

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2008[k,:]= final[j,:]
            k=k+1

```

```

countout2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2009=countout2009+1

```

```

out_2009 = np.zeros((countout2009,8))

```

```

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2009[k,:]= final[j,:]
            k=k+1

```

```

countout2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2010=countout2010+1

```

```

out_2010 = np.zeros((countout2010,8))

```

```

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2010[k,:]= final[j,:]
            k=k+1

```

```

countout2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2011=countout2011+1

```

```

out_2011 = np.zeros((countout2011,8))

```

```

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2011[k,:]= final[j,:]
            k=k+1

```

```

countout2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2012=countout2012+1

```

```

out_2012 = np.zeros((countout2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2012[k,:]= final[j,:]
            k=k+1

countout2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2013=countout2013+1

out_2013 = np.zeros((countout2013,8))

k=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2013[k,:]= final[j,:]
            k=k+1
countout2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2014=countout2014+1

out_2014 = np.zeros((countout2014,8))

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2014[k,:]= final[j,:]
            k=k+1

countout2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2015=countout2015+1

out_2015 = np.zeros((countout2015,8))

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2015[k,:]= final[j,:]
            k=k+1

countout2016=0
for i in range (2016, 2017):

```

```

for j in range (0,sizen):
    if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
        countout2016=countout2016+1

out_2016 = np.zeros((countout2016,8))

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2016[k,:]= final[j,:]
            k=k+1
countout2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            countout2017=countout2017+1

out_2017 = np.zeros((countout2017,8))

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=274 and final[j,2]<305:
            out_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#NOVEMBROS
countnov2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2008=countnov2008+1

nov_2008 = np.zeros((countnov2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2008[k,:]= final[j,:]
            k=k+1

countnov2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2009=countnov2009+1

nov_2009 = np.zeros((countnov2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:

```



```

    nov_2009[k,:]= final[j,:]
    k=k+1

countnov2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2010=countnov2010+1

nov_2010 = np.zeros((countnov2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2010[k,:]= final[j,:]
            k=k+1
countnov2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2011=countnov2011+1

nov_2011 = np.zeros((countnov2011,8))

k=0
for i in range (2011, 2012):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2011[k,:]= final[j,:]
            k=k+1

countnov2012=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2012=countnov2012+1

nov_2012 = np.zeros((countnov2012,8))

k=0
for i in range (2012, 2013):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2012[k,:]= final[j,:]
            k=k+1

countnov2013=0
for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2013=countnov2013+1

nov_2013 = np.zeros((countnov2013,8))

k=0

```

```

for i in range (2013, 2014):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2013[k,:]= final[j,:]
            k=k+1
countnov2014=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2014=countnov2014+1

```

```
nov_2014 = np.zeros((countnov2014,8))
```

```

k=0
for i in range (2014, 2015):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2014[k,:]= final[j,:]
            k=k+1

```

```

countnov2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2015=countnov2015+1

```

```
nov_2015 = np.zeros((countnov2015,8))
```

```

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2015[k,:]= final[j,:]
            k=k+1

```

```

countnov2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2016=countnov2016+1

```

```
nov_2016 = np.zeros((countnov2016,8))
```

```

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2016[k,:]= final[j,:]
            k=k+1
countnov2017=0

```

```

for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            countnov2017=countnov2017+1

```

```
nov_2017 = np.zeros((countnov2017,8))
```

```

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=305 and final[j,2]<335:
            nov_2017[k,:]= final[j,:]
            k=k+1
#####
#####
#DEZEMBROS
countdez2008=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            countdez2008=countdez2008+1

dez_2008 = np.zeros((countdez2008,8))

k=0
for i in range (2008, 2009):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            dez_2008[k,:]= final[j,:]
            k=k+1

countdez2009=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            countdez2009=countdez2009+1

dez_2009 = np.zeros((countdez2009,8))

k=0
for i in range (2009, 2010):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            dez_2009[k,:]= final[j,:]
            k=k+1

countdez2010=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            countdez2010=countdez2010+1

dez_2010 = np.zeros((countdez2010,8))

k=0
for i in range (2010, 2011):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            dez_2010[k,:]= final[j,:]
            k=k+1
countdez2011=0
for i in range (2011, 2012):
    for j in range (0,sizen):

```

```
if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
    countdez2011=countdez2011+1
```

```
dez_2011 = np.zeros((countdez2011,8))
```

```
k=0  
for i in range (2011, 2012):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            dez_2011[k,:]= final[j,:]  
            k=k+1
```

```
countdez2012=0  
for i in range (2012, 2013):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            countdez2012=countdez2012+1
```

```
dez_2012 = np.zeros((countdez2012,8))
```

```
k=0  
for i in range (2012, 2013):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            dez_2012[k,:]= final[j,:]  
            k=k+1
```

```
countdez2013=0  
for i in range (2013, 2014):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            countdez2013=countdez2013+1
```

```
dez_2013 = np.zeros((countdez2013,8))
```

```
k=0  
for i in range (2013, 2014):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            dez_2013[k,:]= final[j,:]  
            k=k+1
```

```
countdez2014=0  
for i in range (2014, 2015):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            countdez2014=countdez2014+1
```

```
dez_2014 = np.zeros((countdez2014,8))
```

```
k=0  
for i in range (2014, 2015):  
    for j in range (0,sizen):  
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:  
            dez_2014[k,:]= final[j,:]  
            k=k+1
```

```

countdez2015=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            countdez2015=countdez2015+1

```

```

dez_2015 = np.zeros((countdez2015,8))

```

```

k=0
for i in range (2015, 2016):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            dez_2015[k,:]= final[j,:]
            k=k+1

```

```

countdez2016=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            countdez2016=countdez2016+1

```

```

dez_2016 = np.zeros((countdez2016,8))

```

```

k=0
for i in range (2016, 2017):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            dez_2016[k,:]= final[j,:]
            k=k+1

```

```

countdez2017=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            countdez2017=countdez2017+1

```

```

dez_2017 = np.zeros((countdez2017,8))

```

```

k=0
for i in range (2017, 2018):
    for j in range (0,sizen):
        if final[j,1] == i and final[j,2] >=335 and final[j,2]<366:
            dez_2017[k,:]= final[j,:]
            k=k+1

```

```

#####

```

```

#####

```

```

#GRAFICOS

```

```

rr = [i for i in range(0,950)]
tt = [i for i in range(50,1000)]

```

```

ss = [i for i in range (0,1050)]
jj = [i for i in range (-50,1000)]

```

```

#JANEIRO

```

```

A=jan_2008[:,4]-jan_2008[:,5]
B=jan_2008[:,6]*jan_2008[:,7]
m=jan_2008[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2008')
plt.scatter(m,K)
plt.savefig('jan2008m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2008d.jpg')
plt.show()

```

```

A=jan_2009[:,4]-jan_2009[:,5]
B=jan_2009[:,6]*jan_2009[:,7]
m=jan_2009[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2009')
plt.scatter(m,K)
plt.savefig('jan2009m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2009d.jpg')
plt.show()

```

```

A=jan_2010[:,4]-jan_2010[:,5]
B=jan_2010[:,6]*jan_2010[:,7]
m=jan_2010[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2010')
plt.scatter(m,K)
plt.savefig('jan2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2010')

```

```

ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2010d.jpg')
plt.show()

```

```

A=jan_2011[:,4]-jan_2011[:,5]
B=jan_2011[:,6]*jan_2011[:,7]
m=jan_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2011')
plt.scatter(m,K)
plt.savefig('jan2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2011d.jpg')
plt.show()

```

```

A=jan_2012[:,4]-jan_2012[:,5]
B=jan_2012[:,6]*jan_2012[:,7]
m=jan_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2012')
plt.scatter(m,K)
plt.savefig('jan2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2012d.jpg')
plt.show()

```

```

A=jan_2013[:,4]-jan_2013[:,5]
B=jan_2013[:,6]*jan_2013[:,7]
m=jan_2013[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2013')
plt.scatter(m,K)
plt.savefig('jan2013m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2013d.jpg')
plt.show()

```

```

A=jan_2014[:,4]-jan_2014[:,5]
B=jan_2014[:,6]*jan_2014[:,7]
m=jan_2014[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2014')
plt.scatter(m,K)
plt.savefig('jan2014m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2014d.jpg')
plt.show()

```

```

A=jan_2015[:,4]-jan_2015[:,5]
B=jan_2015[:,6]*jan_2015[:,7]
m=jan_2015[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2015')
plt.scatter(m,K)
plt.savefig('jan2015m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2015')

```



```

ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2015d.jpg')
plt.show()

```

```

A=jan_2016[:,4]-jan_2016[:,5]
B=jan_2016[:,6]*jan_2016[:,7]
m=jan_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2016')
plt.scatter(m,K)
plt.savefig('jan2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2016d.jpg')
plt.show()

```

```

A=jan_2017[:,4]-jan_2017[:,5]
B=jan_2017[:,6]*jan_2017[:,7]
m=jan_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiro 2017')
plt.scatter(m,K)
plt.savefig('jan2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiro 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jan2017d.jpg')
plt.show()

```

```

#FEVEREIRO
A=fev_2008[:,4]-fev_2008[:,5]
B=fev_2008[:,6]*fev_2008[:,7]

```

```

m=fev_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2008')
plt.scatter(m,K)
plt.savefig('fev2008m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2008d.jpg')
plt.show()

```

```

A=fev_2009[:,4]-fev_2009[:,5]
B=fev_2009[:,6]*fev_2009[:,7]
m=fev_2009[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2009')
plt.scatter(m,K)
plt.savefig('fev2009m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2009d.jpg')
plt.show()

```

```

A=fev_2010[:,4]-fev_2010[:,5]
B=fev_2010[:,6]*fev_2010[:,7]
m=fev_2010[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2010')
plt.scatter(m,K)
plt.savefig('fev2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')

```

```

plt.title('Fevereiro 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2010d.jpg')
plt.show()

```

```

A=fev_2011[:,4]-fev_2011[:,5]
B=fev_2011[:,6]*fev_2011[:,7]
m=fev_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2011')
plt.scatter(m,K)
plt.savefig('fev2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2011d.jpg')
plt.show()

```

```

A=fev_2012[:,4]-fev_2012[:,5]
B=fev_2012[:,6]*fev_2012[:,7]
m=fev_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2012')
plt.scatter(m,K)
plt.savefig('fev2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2012d.jpg')
plt.show()

```

```

A=fev_2013[:,4]-fev_2013[:,5]
B=fev_2013[:,6]*fev_2013[:,7]

```

```

m=fev_2013[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2013')
plt.scatter(m,K)
plt.savefig('fev2013m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2013d.jpg')
plt.show()

```

```

A=fev_2014[:,4]-fev_2014[:,5]
B=fev_2014[:,6]*fev_2014[:,7]
m=fev_2014[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2014')
plt.scatter(m,K)
plt.savefig('fev2014m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2014d.jpg')
plt.show()

```

```

A=fev_2015[:,4]-fev_2015[:,5]
B=fev_2015[:,6]*fev_2015[:,7]
m=fev_2015[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2015')
plt.scatter(m,K)
plt.savefig('fev2015m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')

```

```

plt.title('Fevereiro 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2015d.jpg')
plt.show()

```

```

A=fev_2016[:,4]-fev_2016[:,5]
B=fev_2016[:,6]*fev_2016[:,7]
m=fev_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2016')
plt.scatter(m,K)
plt.savefig('fev2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2016d.jpg')
plt.show()

```

```

A=fev_2017[:,4]-fev_2017[:,5]
B=fev_2017[:,6]*fev_2017[:,7]
m=fev_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Fevereiro 2017')
plt.scatter(m,K)
plt.savefig('fev2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiro 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('fev2017d.jpg')
plt.show()

```

```

#MARCO
A=mar_2008[:,4]-mar_2008[:,5]

```

```

B=mar_2008[:,6]*mar_2008[:,7]
m=mar_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2008')
plt.scatter(m,K)
plt.savefig('mar2008m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2008d.jpg')
plt.show()

```

```

A=mar_2009[:,4]-mar_2009[:,5]
B=mar_2009[:,6]*mar_2009[:,7]
m=mar_2009[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2009')
plt.scatter(m,K)
plt.savefig('mar2009m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2009d.jpg')
plt.show()

```

```

A=mar_2010[:,4]-mar_2010[:,5]
B=mar_2010[:,6]*mar_2010[:,7]
m=mar_2010[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2010')
plt.scatter(m,K)
plt.savefig('mar2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')

```

```

plt.ylabel('dire*cosu')
plt.title('Marco 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2010d.jpg')
plt.show()

```

```

A=mar_2011[:,4]-mar_2011[:,5]
B=mar_2011[:,6]*mar_2011[:,7]
m=mar_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2011')
plt.scatter(m,K)
plt.savefig('mar2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2011d.jpg')
plt.show()

```

```

A=mar_2012[:,4]-mar_2012[:,5]
B=mar_2012[:,6]*mar_2012[:,7]
m=mar_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2012')
plt.scatter(m,K)
plt.savefig('mar2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2012d.jpg')
plt.show()

```

```

A=mar_2013[:,4]-mar_2013[:,5]

```

```

B=mar_2013[:,6]*mar_2013[:,7]
m=mar_2013[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2013')
plt.scatter(m,K)
plt.savefig('mar2013m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2013d.jpg')
plt.show()

```

```

A=mar_2014[:,4]-mar_2014[:,5]
B=mar_2014[:,6]*mar_2014[:,7]
m=mar_2014[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2014')
plt.scatter(m,K)
plt.savefig('mar2014m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2014d.jpg')
plt.show()

```

```

A=mar_2015[:,4]-mar_2015[:,5]
B=mar_2015[:,6]*mar_2015[:,7]
m=mar_2015[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2015')
plt.scatter(m,K)
plt.savefig('mar2015m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')

```



```

plt.ylabel('dire*cosu')
plt.title('Marco 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2015d.jpg')
plt.show()

```

```

A=mar_2016[:,4]-mar_2016[:,5]
B=mar_2016[:,6]*mar_2016[:,7]
m=mar_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2016')
plt.scatter(m,K)
plt.savefig('mar2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2016d.jpg')
plt.show()

```

```

A=mar_2017[:,4]-mar_2017[:,5]
B=mar_2017[:,6]*mar_2017[:,7]
m=mar_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Marco 2017')
plt.scatter(m,K)
plt.savefig('mar2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marco 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mar2017d.jpg')
plt.show()

```

```

#ABRIL

```

```

A=abr_2008[:,4]-abr_2008[:,5]
B=abr_2008[:,6]*abr_2008[:,7]
m=abr_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2008')
plt.scatter(m,K)
plt.savefig('abr2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2008d.jpg')
plt.show()

A=abr_2009[:,4]-abr_2009[:,5]
B=abr_2009[:,6]*abr_2009[:,7]
m=abr_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2009')
plt.scatter(m,K)
plt.savefig('abr2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2009d.jpg')
plt.show()

A=abr_2010[:,4]-abr_2010[:,5]
B=abr_2010[:,6]*abr_2010[:,7]
m=abr_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2010')
plt.scatter(m,K)
plt.savefig('abr2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2010d.jpg')
plt.show()

```

```

A=abr_2011[:,4]-abr_2011[:,5]
B=abr_2011[:,6]*abr_2011[:,7]
m=abr_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2011')
plt.scatter(m,K)
plt.savefig('abr2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2011d.jpg')
plt.show()

```

```

A=abr_2012[:,4]-abr_2012[:,5]
B=abr_2012[:,6]*abr_2012[:,7]
m=abr_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2012')
plt.scatter(m,K)
plt.savefig('abr2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2012d.jpg')
plt.show()

```

```
A=abr_2013[:,4]-abr_2013[:,5]
B=abr_2013[:,6]*abr_2013[:,7]
m=abr_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2013')
plt.scatter(m,K)
plt.savefig('abr2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2013d.jpg')
plt.show()
```

```
A=abr_2014[:,4]-abr_2014[:,5]
B=abr_2014[:,6]*abr_2014[:,7]
m=abr_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2014')
plt.scatter(m,K)
plt.savefig('abr2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2014d.jpg')
plt.show()
```

```
A=abr_2015[:,4]-abr_2015[:,5]
B=abr_2015[:,6]*abr_2015[:,7]
m=abr_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2015')
plt.scatter(m,K)
plt.savefig('abr2015m.jpg')
plt.show()
```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2015d.jpg')
plt.show()

```

```

A=abr_2016[:,4]-abr_2016[:,5]
B=abr_2016[:,6]*abr_2016[:,7]
m=abr_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2016')
plt.scatter(m,K)
plt.savefig('abr2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2016d.jpg')
plt.show()

```

```

A=abr_2017[:,4]-abr_2017[:,5]
B=abr_2017[:,6]*abr_2017[:,7]
m=abr_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Abril 2017')
plt.scatter(m,K)
plt.savefig('abr2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abril 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('abr2017d.jpg')
plt.show()

```

```

#MAIO
A=mai_2008[:,4]-mai_2008[:,5]
B=mai_2008[:,6]*mai_2008[:,7]
m=mai_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2008')
plt.scatter(m,K)
plt.savefig('mai2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2008d.jpg')
plt.show()

A=mai_2009[:,4]-mai_2009[:,5]
B=mai_2009[:,6]*mai_2009[:,7]
m=mai_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2009')
plt.scatter(m,K)
plt.savefig('mai2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2009d.jpg')
plt.show()

A=mai_2010[:,4]-mai_2010[:,5]
B=mai_2010[:,6]*mai_2010[:,7]
m=mai_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2010')
plt.scatter(m,K)
plt.savefig('mai2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2010d.jpg')
plt.show()

```

```

A=mai_2011[:,4]-mai_2011[:,5]
B=mai_2011[:,6]*mai_2011[:,7]
m=mai_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2011')
plt.scatter(m,K)
plt.savefig('mai2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2011d.jpg')
plt.show()

```

```

A=mai_2012[:,4]-mai_2012[:,5]
B=mai_2012[:,6]*mai_2012[:,7]
m=mai_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2012')
plt.scatter(m,K)
plt.savefig('mai2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2012d.jpg')
plt.show()

```

```

A=mai_2013[:,4]-mai_2013[:,5]
B=mai_2013[:,6]*mai_2013[:,7]
m=mai_2013[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2013')
plt.scatter(m,K)
plt.savefig('mai2013m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2013d.jpg')
plt.show()

```

```

A=mai_2014[:,4]-mai_2014[:,5]
B=mai_2014[:,6]*mai_2014[:,7]
m=mai_2014[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2014')
plt.scatter(m,K)
plt.savefig('mai2014m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2014d.jpg')
plt.show()

```

```

A=mai_2015[:,4]-mai_2015[:,5]
B=mai_2015[:,6]*mai_2015[:,7]
m=mai_2015[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2015')
plt.scatter(m,K)
plt.savefig('mai2015m.jpg')
plt.show()

```



```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2015d.jpg')
plt.show()

```

```

A=mai_2016[:,4]-mai_2016[:,5]
B=mai_2016[:,6]*mai_2016[:,7]
m=mai_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2016')
plt.scatter(m,K)
plt.savefig('mai2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2016d.jpg')
plt.show()

```

```

A=mai_2017[:,4]-mai_2017[:,5]
B=mai_2017[:,6]*mai_2017[:,7]
m=mai_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Maio 2017')
plt.scatter(m,K)
plt.savefig('mai2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maio 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('mai2017d.jpg')
plt.show()

```

```

#JUNHO
A=jun_2008[:,4]-jun_2008[:,5]
B=jun_2008[:,6]*jun_2008[:,7]
m=jun_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2008')
plt.scatter(m,K)
plt.savefig('jun2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2008d.jpg')
plt.show()

A=jun_2009[:,4]-jun_2009[:,5]
B=jun_2009[:,6]*jun_2009[:,7]
m=jun_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2009')
plt.scatter(m,K)
plt.savefig('jun2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2009d.jpg')
plt.show()

A=jun_2010[:,4]-jun_2010[:,5]
B=jun_2010[:,6]*jun_2010[:,7]
m=jun_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2010')
plt.scatter(m,K)
plt.savefig('jun2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2010d.jpg')
plt.show()

```

```

A=jun_2011[:,4]-jun_2011[:,5]
B=jun_2011[:,6]*jun_2011[:,7]
m=jun_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2011')
plt.scatter(m,K)
plt.savefig('jun2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2011d.jpg')
plt.show()

```

```

A=jun_2012[:,4]-jun_2012[:,5]
B=jun_2012[:,6]*jun_2012[:,7]
m=jun_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2012')
plt.scatter(m,K)
plt.savefig('jun2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2012d.jpg')
plt.show()

```

```
A=jun_2013[:,4]-jun_2013[:,5]
B=jun_2013[:,6]*jun_2013[:,7]
m=jun_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2013')
plt.scatter(m,K)
plt.savefig('jun2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2013d.jpg')
plt.show()
```

```
A=jun_2014[:,4]-jun_2014[:,5]
B=jun_2014[:,6]*jun_2014[:,7]
m=jun_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2014')
plt.scatter(m,K)
plt.savefig('jun2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2014d.jpg')
plt.show()
```

```
A=jun_2015[:,4]-jun_2015[:,5]
B=jun_2015[:,6]*jun_2015[:,7]
m=jun_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2015')
plt.scatter(m,K)
plt.savefig('jun2015m.jpg')
plt.show()
```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2015d.jpg')
plt.show()

```

```

A=jun_2016[:,4]-jun_2016[:,5]
B=jun_2016[:,6]*jun_2016[:,7]
m=jun_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2016')
plt.scatter(m,K)
plt.savefig('jun2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2016d.jpg')
plt.show()

```

```

A=jun_2017[:,4]-jun_2017[:,5]
B=jun_2017[:,6]*jun_2017[:,7]
m=jun_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Junho 2017')
plt.scatter(m,K)
plt.savefig('jun2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junho 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jun2017d.jpg')
plt.show()

```

```

#JULHO
A=jul_2008[:,4]-jul_2008[:,5]
B=jul_2008[:,6]*jul_2008[:,7]
m=jul_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2008')
plt.scatter(m,K)
plt.savefig('jul2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2008d.jpg')
plt.show()

A=jul_2009[:,4]-jul_2009[:,5]
B=jul_2009[:,6]*jul_2009[:,7]
m=jul_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2009')
plt.scatter(m,K)
plt.savefig('jul2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2009d.jpg')
plt.show()

A=jul_2010[:,4]-jul_2010[:,5]
B=jul_2010[:,6]*jul_2010[:,7]
m=jul_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2010')
plt.scatter(m,K)
plt.savefig('jul2010m.jpg')

```

```

plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2010d.jpg')
plt.show()

```

```

A=jul_2011[:,4]-jul_2011[:,5]
B=jul_2011[:,6]*jul_2011[:,7]
m=jul_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2011')
plt.scatter(m,K)
plt.savefig('jul2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2011d.jpg')
plt.show()

```

```

A=jul_2012[:,4]-jul_2012[:,5]
B=jul_2012[:,6]*jul_2012[:,7]
m=jul_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2012')
plt.scatter(m,K)
plt.savefig('jul2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2012d.jpg')

```

```

plt.show()

A=jul_2013[:,4]-jul_2013[:,5]
B=jul_2013[:,6]*jul_2013[:,7]
m=jul_2013[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2013')
plt.scatter(m,K)
plt.savefig('jul2013m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2013d.jpg')
plt.show()

A=jul_2014[:,4]-jul_2014[:,5]
B=jul_2014[:,6]*jul_2014[:,7]
m=jul_2014[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2014')
plt.scatter(m,K)
plt.savefig('jul2014m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2014d.jpg')
plt.show()

A=jul_2015[:,4]-jul_2015[:,5]
B=jul_2015[:,6]*jul_2015[:,7]
m=jul_2015[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2015')
plt.scatter(m,K)
plt.savefig('jul2015m.jpg')

```



```

plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2015d.jpg')
plt.show()

```

```

A=jul_2016[:,4]-jul_2016[:,5]
B=jul_2016[:,6]*jul_2016[:,7]
m=jul_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2016')
plt.scatter(m,K)
plt.savefig('jul2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2016d.jpg')
plt.show()

```

```

A=jul_2017[:,4]-jul_2017[:,5]
B=jul_2017[:,6]*jul_2017[:,7]
m=jul_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2017')
plt.scatter(m,K)
plt.savefig('jul2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2017d.jpg')

```

```

plt.show()
A=jul_2008[:,4]-jul_2008[:,5]
B=jul_2008[:,6]*jul_2008[:,7]
m=jul_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2008')
plt.scatter(m,K)
plt.savefig('jul2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2008d.jpg')
plt.show()

A=jul_2009[:,4]-jul_2009[:,5]
B=jul_2009[:,6]*jul_2009[:,7]
m=jul_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2009')
plt.scatter(m,K)
plt.savefig('jul2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2009d.jpg')
plt.show()

A=jul_2010[:,4]-jul_2010[:,5]
B=jul_2010[:,6]*jul_2010[:,7]
m=jul_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2010')
plt.scatter(m,K)
plt.savefig('jul2010m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2010d.jpg')
plt.show()

```

```

A=jul_2011[:,4]-jul_2011[:,5]
B=jul_2011[:,6]*jul_2011[:,7]
m=jul_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2011')
plt.scatter(m,K)
plt.savefig('jul2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2011d.jpg')
plt.show()

```

```

A=jul_2012[:,4]-jul_2012[:,5]
B=jul_2012[:,6]*jul_2012[:,7]
m=jul_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2012')
plt.scatter(m,K)
plt.savefig('jul2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2012d.jpg')
plt.show()

```

```
A=jul_2013[:,4]-jul_2013[:,5]
B=jul_2013[:,6]*jul_2013[:,7]
m=jul_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2013')
plt.scatter(m,K)
plt.savefig('jul2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2013d.jpg')
plt.show()
```

```
A=jul_2014[:,4]-jul_2014[:,5]
B=jul_2014[:,6]*jul_2014[:,7]
m=jul_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2014')
plt.scatter(m,K)
plt.savefig('jul2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2014d.jpg')
plt.show()
```

```
A=jul_2015[:,4]-jul_2015[:,5]
B=jul_2015[:,6]*jul_2015[:,7]
m=jul_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2015')
plt.scatter(m,K)
plt.savefig('jul2015m.jpg')
plt.show()
```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2015d.jpg')
plt.show()

```

```

A=jul_2016[:,4]-jul_2016[:,5]
B=jul_2016[:,6]*jul_2016[:,7]
m=jul_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2016')
plt.scatter(m,K)
plt.savefig('jul2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2016d.jpg')
plt.show()

```

```

A=jul_2017[:,4]-jul_2017[:,5]
B=jul_2017[:,6]*jul_2017[:,7]
m=jul_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Julho 2017')
plt.scatter(m,K)
plt.savefig('jul2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julho 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('jul2017d.jpg')
plt.show()

```

```

#AGOSTO
A=ago_2008[:,4]-ago_2008[:,5]
B=ago_2008[:,6]*ago_2008[:,7]
m=ago_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2008')
plt.scatter(m,K)
plt.savefig('ago2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2008d.jpg')
plt.show()

A=ago_2009[:,4]-ago_2009[:,5]
B=ago_2009[:,6]*ago_2009[:,7]
m=ago_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2009')
plt.scatter(m,K)
plt.savefig('ago2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2009d.jpg')
plt.show()

A=ago_2010[:,4]-ago_2010[:,5]
B=ago_2010[:,6]*ago_2010[:,7]
m=ago_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2010')
plt.scatter(m,K)
plt.savefig('ago2010m.jpg')

```

```

plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2010d.jpg')
plt.show()

```

```

A=ago_2011[:,4]-ago_2011[:,5]
B=ago_2011[:,6]*ago_2011[:,7]
m=ago_2011[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2011')
plt.scatter(m,K)
plt.savefig('ago2011m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2011d.jpg')
plt.show()

```

```

A=ago_2012[:,4]-ago_2012[:,5]
B=ago_2012[:,6]*ago_2012[:,7]
m=ago_2012[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2012')
plt.scatter(m,K)
plt.savefig('ago2012m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2012d.jpg')

```

```

plt.show()

A=ago_2013[:,4]-ago_2013[:,5]
B=ago_2013[:,6]*ago_2013[:,7]
m=ago_2013[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2013')
plt.scatter(m,K)
plt.savefig('ago2013m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2013d.jpg')
plt.show()

A=ago_2014[:,4]-ago_2014[:,5]
B=ago_2014[:,6]*ago_2014[:,7]
m=ago_2014[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2014')
plt.scatter(m,K)
plt.savefig('ago2014m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2014d.jpg')
plt.show()

A=ago_2015[:,4]-ago_2015[:,5]
B=ago_2015[:,6]*ago_2015[:,7]
m=ago_2015[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2015')
plt.scatter(m,K)
plt.savefig('ago2015m.jpg')

```



```

plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2015d.jpg')
plt.show()

```

```

A=ago_2016[:,4]-ago_2016[:,5]
B=ago_2016[:,6]*ago_2016[:,7]
m=ago_2016[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2016')
plt.scatter(m,K)
plt.savefig('ago2016m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2016d.jpg')
plt.show()

```

```

A=ago_2017[:,4]-ago_2017[:,5]
B=ago_2017[:,6]*ago_2017[:,7]
m=ago_2017[:,3]

```

```

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Agosto 2017')
plt.scatter(m,K)
plt.savefig('ago2017m.jpg')
plt.show()

```

```

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agosto 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('ago2017d.jpg')

```

```

plt.show()

#SETEMBRO
A=set_2008[:,4]-set_2008[:,5]
B=set_2008[:,6]*set_2008[:,7]
m=set_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2008')
plt.scatter(m,K)
plt.savefig('set2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2008d.jpg')
plt.show()

A=set_2009[:,4]-set_2009[:,5]
B=set_2009[:,6]*set_2009[:,7]
m=set_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2009')
plt.scatter(m,K)
plt.savefig('set2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2009d.jpg')
plt.show()

A=set_2010[:,4]-set_2010[:,5]
B=set_2010[:,6]*set_2010[:,7]
m=set_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2010')
plt.scatter(m,K)

```

```
plt.savefig('set2010m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2010d.jpg')
plt.show()
```

```
A=set_2011[:,4]-set_2011[:,5]
B=set_2011[:,6]*set_2011[:,7]
m=set_2011[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2011')
plt.scatter(m,K)
plt.savefig('set2011m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2011d.jpg')
plt.show()
```

```
A=set_2012[:,4]-set_2012[:,5]
B=set_2012[:,6]*set_2012[:,7]
m=set_2012[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2012')
plt.scatter(m,K)
plt.savefig('set2012m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
```

```
plt.savefig('set2012d.jpg')
plt.show()
```

```
A=set_2013[:,4]-set_2013[:,5]
B=set_2013[:,6]*set_2013[:,7]
m=set_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2013')
plt.scatter(m,K)
plt.savefig('set2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2013d.jpg')
plt.show()
```

```
A=set_2014[:,4]-set_2014[:,5]
B=set_2014[:,6]*set_2014[:,7]
m=set_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2014')
plt.scatter(m,K)
plt.savefig('set2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2014d.jpg')
plt.show()
```

```
A=set_2015[:,4]-set_2015[:,5]
B=set_2015[:,6]*set_2015[:,7]
m=set_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2015')
plt.scatter(m,K)
```

```
plt.savefig('set2015m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2015d.jpg')
plt.show()
```

```
A=set_2016[:,4]-set_2016[:,5]
B=set_2016[:,6]*set_2016[:,7]
m=set_2016[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2016')
plt.scatter(m,K)
plt.savefig('set2016m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('set2016d.jpg')
plt.show()
```

```
A=set_2017[:,4]-set_2017[:,5]
B=set_2017[:,6]*set_2017[:,7]
m=set_2017[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Setembro 2017')
plt.scatter(m,K)
plt.savefig('set2017m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembro 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
```

```

plt.savefig('set2017d.jpg')
plt.show()

#OUTUBRO
A=out_2008[:,4]-out_2008[:,5]
B=out_2008[:,6]*out_2008[:,7]
m=out_2008[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2008')
plt.scatter(m,K)
plt.savefig('out2008m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2008d.jpg')
plt.show()

A=out_2009[:,4]-out_2009[:,5]
B=out_2009[:,6]*out_2009[:,7]
m=out_2009[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2009')
plt.scatter(m,K)
plt.savefig('out2009m.jpg')
plt.show()

plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2009d.jpg')
plt.show()

A=out_2010[:,4]-out_2010[:,5]
B=out_2010[:,6]*out_2010[:,7]
m=out_2010[:,3]

K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2010')

```

```
plt.scatter(m,K)
plt.savefig('out2010m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2010d.jpg')
plt.show()
```

```
A=out_2011[:,4]-out_2011[:,5]
B=out_2011[:,6]*out_2011[:,7]
m=out_2011[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2011')
plt.scatter(m,K)
plt.savefig('out2011m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2011d.jpg')
plt.show()
```

```
A=out_2012[:,4]-out_2012[:,5]
B=out_2012[:,6]*out_2012[:,7]
m=out_2012[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2012')
plt.scatter(m,K)
plt.savefig('out2012m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
```

```
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2012d.jpg')
plt.show()
```

```
A=out_2013[:,4]-out_2013[:,5]
B=out_2013[:,6]*out_2013[:,7]
m=out_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2013')
plt.scatter(m,K)
plt.savefig('out2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2013d.jpg')
plt.show()
```

```
A=out_2014[:,4]-out_2014[:,5]
B=out_2014[:,6]*out_2014[:,7]
m=out_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2014')
plt.scatter(m,K)
plt.savefig('out2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2014d.jpg')
plt.show()
```

```
A=out_2015[:,4]-out_2015[:,5]
B=out_2015[:,6]*out_2015[:,7]
m=out_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2015')
```



```
plt.scatter(m,K)
plt.savefig('out2015m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2015d.jpg')
plt.show()
```

```
A=out_2016[:,4]-out_2016[:,5]
B=out_2016[:,6]*out_2016[:,7]
m=out_2016[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2016')
plt.scatter(m,K)
plt.savefig('out2016m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2016d.jpg')
plt.show()
```

```
A=out_2017[:,4]-out_2017[:,5]
B=out_2017[:,6]*out_2017[:,7]
m=out_2017[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Outubro 2017')
plt.scatter(m,K)
plt.savefig('out2017m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubro 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
```

```
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('out2017d.jpg')
plt.show()
```

```
#NOVEMBRO
```

```
A=nov_2008[:,4]-nov_2008[:,5]
B=nov_2008[:,6]*nov_2008[:,7]
m=nov_2008[:,3]
```

```
K=A-B
```

```
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2008')
plt.scatter(m,K)
plt.savefig('nov2008m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(tr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2008d.jpg')
plt.show()
```

```
A=nov_2009[:,4]-nov_2009[:,5]
B=nov_2009[:,6]*nov_2009[:,7]
m=nov_2009[:,3]
```

```
K=A-B
```

```
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2009')
plt.scatter(m,K)
plt.savefig('nov2009m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(tr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2009d.jpg')
plt.show()
```

```
A=nov_2010[:,4]-nov_2010[:,5]
B=nov_2010[:,6]*nov_2010[:,7]
m=nov_2010[:,3]
```

```
K=A-B
```

```
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
```

```
plt.title('Novembro 2010')
plt.scatter(m,K)
plt.savefig('nov2010m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(tr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2010d.jpg')
plt.show()
```

```
A=nov_2011[:,4]-nov_2011[:,5]
B=nov_2011[:,6]*nov_2011[:,7]
m=nov_2011[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2011')
plt.scatter(m,K)
plt.savefig('nov2011m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(tr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2011d.jpg')
plt.show()
```

```
A=nov_2012[:,4]-nov_2012[:,5]
B=nov_2012[:,6]*nov_2012[:,7]
m=nov_2012[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2012')
plt.scatter(m,K)
plt.savefig('nov2012m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
```

```
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2012d.jpg')
plt.show()
```

```
A=nov_2013[:,4]-nov_2013[:,5]
B=nov_2013[:,6]*nov_2013[:,7]
m=nov_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2013')
plt.scatter(m,K)
plt.savefig('nov2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2013d.jpg')
plt.show()
```

```
A=nov_2014[:,4]-nov_2014[:,5]
B=nov_2014[:,6]*nov_2014[:,7]
m=nov_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2014')
plt.scatter(m,K)
plt.savefig('nov2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2014d.jpg')
plt.show()
```

```
A=nov_2015[:,4]-nov_2015[:,5]
B=nov_2015[:,6]*nov_2015[:,7]
m=nov_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
```

```
plt.title('Novembro 2015')
plt.scatter(m,K)
plt.savefig('nov2015m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(tr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2015d.jpg')
plt.show()
```

```
A=nov_2016[:,4]-nov_2016[:,5]
B=nov_2016[:,6]*nov_2016[:,7]
m=nov_2016[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2016')
plt.scatter(m,K)
plt.savefig('nov2016m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(tr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2016d.jpg')
plt.show()
```

```
A=nov_2017[:,4]-nov_2017[:,5]
B=nov_2017[:,6]*nov_2017[:,7]
m=nov_2017[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Novembro 2017')
plt.scatter(m,K)
plt.savefig('nov2017m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembro 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
```

```
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('nov2017d.jpg')
plt.show()
```

```
#DEZEMBRO
```

```
A=dez_2008[:,4]-dez_2008[:,5]
B=dez_2008[:,6]*dez_2008[:,7]
m=dez_2008[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2008')
plt.scatter(m,K)
plt.savefig('dez2008m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2008')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2008d.jpg')
plt.show()
```

```
A=dez_2009[:,4]-dez_2009[:,5]
B=dez_2009[:,6]*dez_2009[:,7]
m=dez_2009[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2009')
plt.scatter(m,K)
plt.savefig('dez2009m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2009')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2009d.jpg')
plt.show()
```

```
A=dez_2010[:,4]-dez_2010[:,5]
B=dez_2010[:,6]*dez_2010[:,7]
m=dez_2010[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
```

```
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2010')
plt.scatter(m,K)
plt.savefig('dez2010m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2010')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2010d.jpg')
plt.show()
```

```
A=dez_2011[:,4]-dez_2011[:,5]
B=dez_2011[:,6]*dez_2011[:,7]
m=dez_2011[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2011')
plt.scatter(m,K)
plt.savefig('dez2011m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2011')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2011d.jpg')
plt.show()
```

```
A=dez_2012[:,4]-dez_2012[:,5]
B=dez_2012[:,6]*dez_2012[:,7]
m=dez_2012[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2012')
plt.scatter(m,K)
plt.savefig('dez2012m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2012')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
```

```
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2012d.jpg')
plt.show()
```

```
A=dez_2013[:,4]-dez_2013[:,5]
B=dez_2013[:,6]*dez_2013[:,7]
m=dez_2013[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2013')
plt.scatter(m,K)
plt.savefig('dez2013m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2013')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2013d.jpg')
plt.show()
```

```
A=dez_2014[:,4]-dez_2014[:,5]
B=dez_2014[:,6]*dez_2014[:,7]
m=dez_2014[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2014')
plt.scatter(m,K)
plt.savefig('dez2014m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2014')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2014d.jpg')
plt.show()
```

```
A=dez_2015[:,4]-dez_2015[:,5]
B=dez_2015[:,6]*dez_2015[:,7]
m=dez_2015[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
```



```
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2015')
plt.scatter(m,K)
plt.savefig('dez2015m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2015')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2015d.jpg')
plt.show()
```

```
A=dez_2016[:,4]-dez_2016[:,5]
B=dez_2016[:,6]*dez_2016[:,7]
m=dez_2016[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2016')
plt.scatter(m,K)
plt.savefig('dez2016m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2016')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2016d.jpg')
plt.show()
```

```
A=dez_2017[:,4]-dez_2017[:,5]
B=dez_2017[:,6]*dez_2017[:,7]
m=dez_2017[:,3]
```

```
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Dezembro 2017')
plt.scatter(m,K)
plt.savefig('dez2017m.jpg')
plt.show()
```

```
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembro 2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
```

```

plt.scatter(A,B)
plt.plot(rr,tt, color = 'k', linewidth = 4.0)
plt.plot(ss,jj, color = 'k', linewidth = 4.0)
plt.savefig('dez2017d.jpg')
plt.show()
#####
####
#GERAL - JANEIROS
a1=jan_2008[:,4]-jan_2008[:,5]
a2=jan_2008[:,6]-jan_2008[:,7]
b1=jan_2009[:,4]-jan_2009[:,5]
b2=jan_2009[:,6]-jan_2009[:,7]
c1=jan_2010[:,4]-jan_2010[:,5]
c2=jan_2010[:,6]-jan_2010[:,7]
d1=jan_2011[:,4]-jan_2011[:,5]
d2=jan_2011[:,6]-jan_2011[:,7]
e1=jan_2012[:,4]-jan_2012[:,5]
e2=jan_2012[:,6]-jan_2012[:,7]
f1=jan_2013[:,4]-jan_2013[:,5]
f2=jan_2013[:,6]-jan_2013[:,7]
g1=jan_2014[:,4]-jan_2014[:,5]
g2=jan_2014[:,6]-jan_2014[:,7]
h1=jan_2015[:,4]-jan_2015[:,5]
h2=jan_2015[:,6]-jan_2015[:,7]
i1=jan_2016[:,4]-jan_2016[:,5]
i2=jan_2016[:,6]-jan_2016[:,7]
j1=jan_2017[:,4]-jan_2017[:,5]
j2=jan_2017[:,6]-jan_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Janeiros 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs', markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms', markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx', markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx', markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('janeiros2008_2017.jpg')
plt.show()
#####
####
#GERAL - FEVEREIRO
a1=fev_2008[:,4]-fev_2008[:,5]
a2=fev_2008[:,6]-fev_2008[:,7]
b1=fev_2009[:,4]-fev_2009[:,5]
b2=fev_2009[:,6]-fev_2009[:,7]
c1=fev_2010[:,4]-fev_2010[:,5]
c2=fev_2010[:,6]-fev_2010[:,7]
d1=fev_2011[:,4]-fev_2011[:,5]
d2=fev_2011[:,6]-fev_2011[:,7]

```

```

e1=fev_2012[:,4]-fev_2012[:,5]
e2=fev_2012[:,6]-fev_2012[:,7]
f1=fev_2013[:,4]-fev_2013[:,5]
f2=fev_2013[:,6]-fev_2013[:,7]
g1=fev_2014[:,4]-fev_2014[:,5]
g2=fev_2014[:,6]-fev_2014[:,7]
h1=fev_2015[:,4]-fev_2015[:,5]
h2=fev_2015[:,6]-fev_2015[:,7]
i1=fev_2016[:,4]-fev_2016[:,5]
i2=fev_2016[:,6]-fev_2016[:,7]
j1=fev_2017[:,4]-fev_2017[:,5]
j2=fev_2017[:,6]-fev_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Fevereiroiros 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs',markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms',markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx',markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx',markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('fevereiros2008_2017.jpg')
plt.show()
#####
#####
#GERAL - MARCOS
a1=mar_2008[:,4]-mar_2008[:,5]
a2=mar_2008[:,6]-mar_2008[:,7]
b1=mar_2009[:,4]-mar_2009[:,5]
b2=mar_2009[:,6]-mar_2009[:,7]
c1=mar_2010[:,4]-mar_2010[:,5]
c2=mar_2010[:,6]-mar_2010[:,7]
d1=mar_2011[:,4]-mar_2011[:,5]
d2=mar_2011[:,6]-mar_2011[:,7]
e1=mar_2012[:,4]-mar_2012[:,5]
e2=mar_2012[:,6]-mar_2012[:,7]
f1=mar_2013[:,4]-mar_2013[:,5]
f2=mar_2013[:,6]-mar_2013[:,7]
g1=mar_2014[:,4]-mar_2014[:,5]
g2=mar_2014[:,6]-mar_2014[:,7]
h1=mar_2015[:,4]-mar_2015[:,5]
h2=mar_2015[:,6]-mar_2015[:,7]
i1=mar_2016[:,4]-mar_2016[:,5]
i2=mar_2016[:,6]-mar_2016[:,7]
j1=mar_2017[:,4]-mar_2017[:,5]
j2=mar_2017[:,6]-mar_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Marcos 2008-2017')
ax=plt.gca()

```

```

ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs', markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms', markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx', markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx', markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('marcos2008_2017.jpg')
plt.show()
#####
#####
#GERAL - ABRIS
a1=abr_2008[:,4]-abr_2008[:,5]
a2=abr_2008[:,6]-abr_2008[:,7]
b1=abr_2009[:,4]-abr_2009[:,5]
b2=abr_2009[:,6]-abr_2009[:,7]
c1=abr_2010[:,4]-abr_2010[:,5]
c2=abr_2010[:,6]-abr_2010[:,7]
d1=abr_2011[:,4]-abr_2011[:,5]
d2=abr_2011[:,6]-abr_2011[:,7]
e1=abr_2012[:,4]-abr_2012[:,5]
e2=abr_2012[:,6]-abr_2012[:,7]
f1=abr_2013[:,4]-abr_2013[:,5]
f2=abr_2013[:,6]-abr_2013[:,7]
g1=abr_2014[:,4]-abr_2014[:,5]
g2=abr_2014[:,6]-abr_2014[:,7]
h1=abr_2015[:,4]-abr_2015[:,5]
h2=abr_2015[:,6]-abr_2015[:,7]
i1=abr_2016[:,4]-abr_2016[:,5]
i2=abr_2016[:,6]-abr_2016[:,7]
j1=abr_2017[:,4]-abr_2017[:,5]
j2=abr_2017[:,6]-abr_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Abris 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs', markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms', markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx', markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx', markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('abris2008_2017.jpg')

```

```

plt.show()
#####
#####
#GERAL - MAIOS
a1=mai_2008[:,4]-mai_2008[:,5]
a2=mai_2008[:,6]-mai_2008[:,7]
b1=mai_2009[:,4]-mai_2009[:,5]
b2=mai_2009[:,6]-mai_2009[:,7]
c1=mai_2010[:,4]-mai_2010[:,5]
c2=mai_2010[:,6]-mai_2010[:,7]
d1=mai_2011[:,4]-mai_2011[:,5]
d2=mai_2011[:,6]-mai_2011[:,7]
e1=mai_2012[:,4]-mai_2012[:,5]
e2=mai_2012[:,6]-mai_2012[:,7]
f1=mai_2013[:,4]-mai_2013[:,5]
f2=mai_2013[:,6]-mai_2013[:,7]
g1=mai_2014[:,4]-mai_2014[:,5]
g2=mai_2014[:,6]-mai_2014[:,7]
h1=mai_2015[:,4]-mai_2015[:,5]
h2=mai_2015[:,6]-mai_2015[:,7]
i1=mai_2016[:,4]-mai_2016[:,5]
i2=mai_2016[:,6]-mai_2016[:,7]
j1=mai_2017[:,4]-mai_2017[:,5]
j2=mai_2017[:,6]-mai_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Maios 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs', markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms', markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx', markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx', markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('maios2008_2017.jpg')
plt.show()
#####
#####
#GERAL - JUNHOS
a1=jun_2008[:,4]-jun_2008[:,5]
a2=jun_2008[:,6]-jun_2008[:,7]
b1=jun_2009[:,4]-jun_2009[:,5]
b2=jun_2009[:,6]-jun_2009[:,7]
c1=jun_2010[:,4]-jun_2010[:,5]
c2=jun_2010[:,6]-jun_2010[:,7]
d1=jun_2011[:,4]-jun_2011[:,5]
d2=jun_2011[:,6]-jun_2011[:,7]
e1=jun_2012[:,4]-jun_2012[:,5]
e2=jun_2012[:,6]-jun_2012[:,7]
f1=jun_2013[:,4]-jun_2013[:,5]
f2=jun_2013[:,6]-jun_2013[:,7]

```

```

g1=jun_2014[:,4]-jun_2014[:,5]
g2=jun_2014[:,6]-jun_2014[:,7]
h1=jun_2015[:,4]-jun_2015[:,5]
h2=jun_2015[:,6]-jun_2015[:,7]
i1=jun_2016[:,4]-jun_2016[:,5]
i2=jun_2016[:,6]-jun_2016[:,7]
j1=jun_2017[:,4]-jun_2017[:,5]
j2=jun_2017[:,6]-jun_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Junhos 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs', markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms', markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx', markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx', markersize=0.5, label='2017')
plt.legend()
plt.plot(tr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('junhos2008_2017.jpg')
plt.show()
#####
#####
#GERAL - JULHOS
a1=jul_2008[:,4]-jul_2008[:,5]
a2=jul_2008[:,6]-jul_2008[:,7]
b1=jul_2009[:,4]-jul_2009[:,5]
b2=jul_2009[:,6]-jul_2009[:,7]
c1=jul_2010[:,4]-jul_2010[:,5]
c2=jul_2010[:,6]-jul_2010[:,7]
d1=jul_2011[:,4]-jul_2011[:,5]
d2=jul_2011[:,6]-jul_2011[:,7]
e1=jul_2012[:,4]-jul_2012[:,5]
e2=jul_2012[:,6]-jul_2012[:,7]
f1=jul_2013[:,4]-jul_2013[:,5]
f2=jul_2013[:,6]-jul_2013[:,7]
g1=jul_2014[:,4]-jul_2014[:,5]
g2=jul_2014[:,6]-jul_2014[:,7]
h1=jul_2015[:,4]-jul_2015[:,5]
h2=jul_2015[:,6]-jul_2015[:,7]
i1=jul_2016[:,4]-jul_2016[:,5]
i2=jul_2016[:,6]-jul_2016[:,7]
j1=jul_2017[:,4]-jul_2017[:,5]
j2=jul_2017[:,6]-jul_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Julhos 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')

```

```

plt.plot(c1,c2, 'gs',markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms',markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx',markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx',markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('julhos2008_2017.jpg')
plt.show()
#####
#####
#GERAL - AGOSTOS
a1=ago_2008[:,4]-ago_2008[:,5]
a2=ago_2008[:,6]-ago_2008[:,7]
b1=ago_2009[:,4]-ago_2009[:,5]
b2=ago_2009[:,6]-ago_2009[:,7]
c1=ago_2010[:,4]-ago_2010[:,5]
c2=ago_2010[:,6]-ago_2010[:,7]
d1=ago_2011[:,4]-ago_2011[:,5]
d2=ago_2011[:,6]-ago_2011[:,7]
e1=ago_2012[:,4]-ago_2012[:,5]
e2=ago_2012[:,6]-ago_2012[:,7]
f1=ago_2013[:,4]-ago_2013[:,5]
f2=ago_2013[:,6]-ago_2013[:,7]
g1=ago_2014[:,4]-ago_2014[:,5]
g2=ago_2014[:,6]-ago_2014[:,7]
h1=ago_2015[:,4]-ago_2015[:,5]
h2=ago_2015[:,6]-ago_2015[:,7]
i1=ago_2016[:,4]-ago_2016[:,5]
i2=ago_2016[:,6]-ago_2016[:,7]
j1=ago_2017[:,4]-ago_2017[:,5]
j2=ago_2017[:,6]-ago_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Agostos 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs',markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms',markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx',markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx',markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('agostos2008_2017.jpg')
plt.show()
#####
#####
#GERAL - SETEMBROS

```

```

a1=set_2008[:,4]-set_2008[:,5]
a2=set_2008[:,6]-set_2008[:,7]
b1=set_2009[:,4]-set_2009[:,5]
b2=set_2009[:,6]-set_2009[:,7]
c1=set_2010[:,4]-set_2010[:,5]
c2=set_2010[:,6]-set_2010[:,7]
d1=set_2011[:,4]-set_2011[:,5]
d2=set_2011[:,6]-set_2011[:,7]
e1=set_2012[:,4]-set_2012[:,5]
e2=set_2012[:,6]-set_2012[:,7]
f1=set_2013[:,4]-set_2013[:,5]
f2=set_2013[:,6]-set_2013[:,7]
g1=set_2014[:,4]-set_2014[:,5]
g2=set_2014[:,6]-set_2014[:,7]
h1=set_2015[:,4]-set_2015[:,5]
h2=set_2015[:,6]-set_2015[:,7]
i1=set_2016[:,4]-set_2016[:,5]
i2=set_2016[:,6]-set_2016[:,7]
j1=set_2017[:,4]-set_2017[:,5]
j2=set_2017[:,6]-set_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Setembros 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs', markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms', markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx', markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx', markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('setembros2008_2017.jpg')
plt.show()
#####
#####
#GERAL - OUTUBROS
a1=out_2008[:,4]-out_2008[:,5]
a2=out_2008[:,6]-out_2008[:,7]
b1=out_2009[:,4]-out_2009[:,5]
b2=out_2009[:,6]-out_2009[:,7]
c1=out_2010[:,4]-out_2010[:,5]
c2=out_2010[:,6]-out_2010[:,7]
d1=out_2011[:,4]-out_2011[:,5]
d2=out_2011[:,6]-out_2011[:,7]
e1=out_2012[:,4]-out_2012[:,5]
e2=out_2012[:,6]-out_2012[:,7]
f1=out_2013[:,4]-out_2013[:,5]
f2=out_2013[:,6]-out_2013[:,7]
g1=out_2014[:,4]-out_2014[:,5]
g2=out_2014[:,6]-out_2014[:,7]
h1=out_2015[:,4]-out_2015[:,5]
h2=out_2015[:,6]-out_2015[:,7]

```



```

i1=out_2016[:,4]-out_2016[:,5]
i2=out_2016[:,6]-out_2016[:,7]
j1=out_2017[:,4]-out_2017[:,5]
j2=out_2017[:,6]-out_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Outubros 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs',markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms',markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx',markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx',markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('outubros2008_2017.jpg')
plt.show()
#####
#####
#GERAL - NOVEMBROS
a1=nov_2008[:,4]-nov_2008[:,5]
a2=nov_2008[:,6]-nov_2008[:,7]
b1=nov_2009[:,4]-nov_2009[:,5]
b2=nov_2009[:,6]-nov_2009[:,7]
c1=nov_2010[:,4]-nov_2010[:,5]
c2=nov_2010[:,6]-nov_2010[:,7]
d1=nov_2011[:,4]-nov_2011[:,5]
d2=nov_2011[:,6]-nov_2011[:,7]
e1=nov_2012[:,4]-nov_2012[:,5]
e2=nov_2012[:,6]-nov_2012[:,7]
f1=nov_2013[:,4]-nov_2013[:,5]
f2=nov_2013[:,6]-nov_2013[:,7]
g1=nov_2014[:,4]-nov_2014[:,5]
g2=nov_2014[:,6]-nov_2014[:,7]
h1=nov_2015[:,4]-nov_2015[:,5]
h2=nov_2015[:,6]-nov_2015[:,7]
i1=nov_2016[:,4]-nov_2016[:,5]
i2=nov_2016[:,6]-nov_2016[:,7]
j1=nov_2017[:,4]-nov_2017[:,5]
j2=nov_2017[:,6]-nov_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Novembros 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs',markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms',markersize=0.5, label='2013')

```

```

plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx',markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx',markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('novembros2008_2017.jpg')
plt.show()
#####
####
#GERAL - DEZEMBROS
a1=dez_2008[:,4]-dez_2008[:,5]
a2=dez_2008[:,6]-dez_2008[:,7]
b1=dez_2009[:,4]-dez_2009[:,5]
b2=dez_2009[:,6]-dez_2009[:,7]
c1=dez_2010[:,4]-dez_2010[:,5]
c2=dez_2010[:,6]-dez_2010[:,7]
d1=dez_2011[:,4]-dez_2011[:,5]
d2=dez_2011[:,6]-dez_2011[:,7]
e1=dez_2012[:,4]-dez_2012[:,5]
e2=dez_2012[:,6]-dez_2012[:,7]
f1=dez_2013[:,4]-dez_2013[:,5]
f2=dez_2013[:,6]-dez_2013[:,7]
g1=dez_2014[:,4]-dez_2014[:,5]
g2=dez_2014[:,6]-dez_2014[:,7]
h1=dez_2015[:,4]-dez_2015[:,5]
h2=dez_2015[:,6]-dez_2015[:,7]
i1=dez_2016[:,4]-dez_2016[:,5]
i2=dez_2016[:,6]-dez_2016[:,7]
j1=dez_2017[:,4]-dez_2017[:,5]
j2=dez_2017[:,6]-dez_2017[:,7]
plt.xlabel('gl-difu')
plt.ylabel('dire*cosu')
plt.title('Dezembros 2008-2017')
ax=plt.gca()
ax.set_ylim([0,1000])
ax.set_xlim([0,1000])
plt.plot(a1,a2, 'bs', markersize=2, label='2008')
plt.plot(b1,b2, 'ys', markersize=1, label='2009')
plt.plot(c1,c2, 'gs',markersize=0.5, label='2010')
plt.plot(d1,d2, 'rs', markersize=2, label='2011')
plt.plot(e1,e2, 'cs', markersize=1, label='2012')
plt.plot(f1,f2, 'ms',markersize=0.5, label='2013')
plt.plot(g1,g2, 'ks', markersize=2, label='2014')
plt.plot(h1,h2, 'bx', markersize=1, label='2015')
plt.plot(i1,i2, 'kx',markersize=0.5, label='2016')
plt.plot(j1,j2, 'rx',markersize=0.5, label='2017')
plt.legend()
plt.plot(rr,tt, color = 'k', linewidth = 3.0)
plt.plot(ss,jj, color = 'k', linewidth = 3.0)
plt.savefig('dezembros2008_2017.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0

```

```

hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#JANEIROS - ANGULO >75 POR MINUTO
dia=janeiros[:,2]
minuto=janeiros[:,3]
gl=janeiros[:,4]
dire=janeiros[:,6]
difu=janeiros[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795

#PRECISA COLOCAR COS U E AJUSTAR OS INDICES DAS MATRIZ MATRIZ[:,7]
final1=np.append(janeiros,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)
size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('Janeiros filtrado')
plt.scatter(C,K)
plt.savefig('janeirosfiltrado.jpg')
plt.show()

```

```

#####
#####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#fevereiros - ANGULO >75 POR MINUTO
dia=fevereiros[:,2]
minuto=fevereiros[:,3]
gl=fevereiros[:,4]
dire=fevereiros[:,6]
difu=fevereiros[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(fevereiros,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)
size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')

```

```

plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('fevereiros filtrado')
plt.scatter(C,K)
plt.savefig('fevereirosfiltrado.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#marcos - ANGULO >75 POR MINUTO
dia=marcos[:,2]
minuto=marcos[:,3]
gl=marcos[:,4]
dire=marcos[:,6]
difu=marcos[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(marcos,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:

```

```

matriz[k,:]= final[i,:]
k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('marcos filtrado')
plt.scatter(C,K)
plt.savefig('marcosfiltrado.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#abris - ANGULO >75 POR MINUTO
dia=abris[:,2]
minuto=abris[:,3]
gl=abris[:,4]
dire=abris[:,6]
difu=abris[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(abris,arctgdegree[:,None], axis=1)

final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

```

```

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('abris filtrado')
plt.scatter(C,K)
plt.savefig('abrisfiltrado.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#maios - ANGULO >75 POR MINUTO
dia=maios[:,2]
minuto=maios[:,3]
gl=maios[:,4]
dire=maios[:,6]
difu=maios[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)

```

```

arctgdegree = arct*57.2957795
final1=np.append(maios,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('maios filtrado')
plt.scatter(C,K)
plt.savefig('maiosfiltrado.jpg')
plt.show()
#####
#####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final1=0
final=0
#junhos - ANGULO >75 POR MINUTO
dia=junhos[:,2]
minuto=junhos[:,3]
gl=junhos[:,4]
dire=junhos[:,6]
dif=junhos[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)

```



```

Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(julhos,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('junhos filtrado')
plt.scatter(C,K)
plt.savefig('junhosfiltrado.jpg')
plt.show()
#####
#####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#julhos - ANGULO >75 POR MINUTO
dia=julhos[:,2]
minuto=julhos[:,3]
gl=julhos[:,4]
dire=julhos[:,6]
dif=julhos[:,5]
angdia=(2*math.pi)*((dia-1)/365)

```

```

Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(julhos,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('julhos filtrado')
plt.scatter(C,K)
plt.savefig('julhosfiltrado.jpg')
plt.show()
#####
#####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#agostos - ANGULO >75 POR MINUTO

```

```

dia=agostos[:,2]
minuto=agostos[:,3]
gl=agostos[:,4]
dire=agostos[:,6]
difu=agostos[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final=np.append(agostos,arctgdegree[:,None], axis=1)
final=np.append(final, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('agostos filtrado')
plt.scatter(C,K)
plt.savefig('agostosfiltrado.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0

```

```

cosu=0
arct=0
arctgdegree=0
final1=0
final=0
#setembros - ANGULO >75 POR MINUTO
dia=setembros[:,2]
minuto=setembros[:,3]
gl=setembros[:,4]
dire=setembros[:,6]
difu=setembros[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(setembros,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('setembros filtrado')
plt.scatter(C,K)
plt.savefig('setembrosfiltrado.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0

```

```

Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final1=0
final=0
#outubros - ANGULO >75 POR MINUTO
dia=outubros[:,2]
minuto=outubros[:,3]
gl=outubros[:,4]
dire=outubros[:,6]
difu=outubros[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(outubros,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('outubros filtrado')
plt.scatter(C,K)
plt.savefig('outubrosfiltrado.jpg')
plt.show()

```

```

#####
#####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#novembros - ANGULO >75 POR MINUTO
dia=novembros[:,2]
minuto=novembros[:,3]
gl=novembros[:,4]
dire=novembros[:,6]
difu=novembros[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(novembros,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]
        k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B

```

```

plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('novembros filtrado')
plt.scatter(C,K)
plt.savefig('novembrosfiltrado.jpg')
plt.show()
#####
####
angdia=0
Eo=0
Sa=0
Dec=0
Et=0
hc=0
w=0
radianodec=0
radianolat=0
radianow=0
cosu=0
arct=0
arctgdegree=0
final=0
final1=0
#dezembros - ANGULO >75 POR MINUTO
dia=dezembros[:,2]
minuto=dezembros[:,3]
gl=dezembros[:,4]
dire=dezembros[:,6]
difu=dezembros[:,5]
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
arct= np.arccos(cosu)
arctgdegree = arct*57.2957795
final1=np.append(dezembros,arctgdegree[:,None], axis=1)
final=np.append(final1, cosu[:,None], axis=1)

size=len(final)

countj=0
for i in range (0,size):
    if final[i,7] <= 75:
        countj=countj+1
matriz = np.zeros((countj,9))
k=0
for i in range (0,size):
    if final[i,7] <= 75:
        matriz[k,:]= final[i,:]

```

```
k=k+1

A=matriz[:,4]-matriz[:,5]
B=matriz[:,6]*matriz[:,8]
C=matriz[:,3]
K=A-B
plt.xlabel('Minuto')
plt.ylabel('Diferença (gl-dif)-(dir*cosu)')
plt.title('dezembros filtrado')
plt.scatter(C,K)
plt.savefig('dezembrosfiltrado.jpg')
plt.show()
```

APÊNDICE B – CÓDIGO DE ELABORAÇÃO DOS HISTOGRAMAS

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import math as math

x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BR
B_2005.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2005')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2005.jpg')
plt.show()

size=len(janz)

count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2005")
print("entre 0 e 25")
```

```

print(count1/size)

count2=0
for j in range (0,size):
    if janz[j] >=26 and janz[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janz[j] >=51 and janz[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janz[j] >=76 and janz[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if janz[j] >=101 and janz[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if janz[j] >=126 and janz[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if janz[j] < 0 and janz[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

```

```

count10=0
for j in range (0,size):
    if jan[z[j] <=-26 and jan[z[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if jan[z[j] <=-51 and jan[z[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if jan[z[j] <=-76 and jan[z[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if jan[z[j] <=-101 and jan[z[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if jan[z[j] <=-126 and jan[z[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if jan[z[j] <=-151 and jan[z[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if jan[z[j] <=-176 and jan[z[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)

```

```

#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
06.txt', delimiter = ',', dtype=float)

```

```

df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.6008333333333
lon=-47.7130555555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2006')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2006.jpg')
plt.show()

size=len(janz)
count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2006")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janb[j] >=26 and janb[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0

```

```

for j in range (0,size):
    if jan[z[j] >=51 and jan[z[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if jan[z[j] >=76 and jan[z[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if jan[z[j] >=101 and jan[z[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if jan[z[j] >=126 and jan[z[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if jan[z[j] >=151 and jan[z[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if jan[z[j] >=176 and jan[z[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if jan[z[j] < 0 and jan[z[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if jan[z[j] <=-26 and jan[z[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0

```

```

for j in range (0,size):
    if jan[z[j] <=-51 and jan[z[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if jan[z[j] <=-76 and jan[z[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if jan[z[j] <=-101 and jan[z[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if jan[z[j] <=-126 and jan[z[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if jan[z[j] <=-151 and jan[z[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if jan[z[j] <=-176 and jan[z[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)

```

```

#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
07.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)

```

```

lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2007')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2007.jpg')
plt.show()

size=len(janz)
count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2007")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janb[j] >=26 and janb[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janb[j] >=51 and janb[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0

```

```

for j in range (0,size):
    if jan[z[j] >=76 and jan[z[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if jan[z[j] >=101 and jan[z[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if jan[z[j] >=126 and jan[z[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if jan[z[j] >=151 and jan[z[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if jan[z[j] >=176 and jan[z[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if jan[z[j] < 0 and jan[z[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if jan[z[j] <=-26 and jan[z[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if jan[z[j] <=-51 and jan[z[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

```



```

count12=0
for j in range (0,size):
    if janz[j] <=-76 and janz[j] >=-100:
        count12 = count12+1

```

```

print("entre -76 e -100")
print(count12/size)

```

```

count13=0
for j in range (0,size):
    if janz[j] <=-101 and janz[j] >=-125:
        count13 = count13+1

```

```

print("entre -101 e -125")
print(count13/size)

```

```

count14=0
for j in range (0,size):
    if janz[j] <=-126 and janz[j] >=-150:
        count14 = count14+1

```

```

print("entre -126 e -150")
print(count14/size)

```

```

count15=0
for j in range (0,size):
    if janz[j] <=-151 and janz[j] >=-175:
        count15 = count15+1

```

```

print("entre -151 e -175")
print(count15/size)

```

```

count16=0
for j in range (0,size):
    if janz[j] <=-176 and janz[j] >=-200:
        count16 = count16+1

```

```

print("entre -176 e -200")
print(count16/size)

```

```

#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
08.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)

```

```

Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2008')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2008.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2008")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janb[j] >=26 and janb[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janb[j] >=51 and janb[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janb[j] >=76 and janb[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if janb[j] >=101 and janb[j] <=125:
        count5 = count5+1

```

```

print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if janz[j] >=126 and janz[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if janz[j] < 0 and janz[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if janz[j] <=-26 and janz[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if janz[j] <=-51 and janz[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if janz[j] <=-76 and janz[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):

```

```

        if jan[z[j] <=-101 and jan[z[j] >=-125:
            count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if jan[z[j] <=-126 and jan[z[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if jan[z[j] <=-151 and jan[z[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if jan[z[j] <=-176 and jan[z[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)

#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
09.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180

```

```

radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2009')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2009.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2009")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janb[j] >=26 and janb[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janb[j] >=51 and janb[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janb[j] >=76 and janb[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if janb[j] >=101 and janb[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if janb[j] >=126 and janb[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0

```

```

for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if janz[j] < 0 and janz[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if janz[j] <=-26 and janz[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if janz[j] <=-51 and janz[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if janz[j] <=-76 and janz[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if janz[j] <=-101 and janz[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if janz[j] <=-126 and janz[j] >=-150:
        count14 = count14+1

```

```

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if janz[j] <=-151 and janz[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if janz[j] <=-176 and janz[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)

#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
10.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.6008333333333
lon=-47.7130555555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=janz-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')

```

```

plt.title('Brasilia 2010')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2010.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janz[j] >=0 and janz[j] <=25:
        count1 = count1+1
print("2010")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janz[j] >=26 and janz[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janz[j] >=51 and janz[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janz[j] >=76 and janz[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if janz[j] >=101 and janz[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if janz[j] >=126 and janz[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1

```



```

print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if janz[j] < 0 and janz[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if janz[j] <=-26 and janz[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if janz[j] <=-51 and janz[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if janz[j] <=-76 and janz[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if janz[j] <=-101 and janz[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if janz[j] <=-126 and janz[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if janz[j] <=-151 and janz[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

```

```

count16=0
for j in range (0,size):
    if janzz[j] <=-176 and janzz[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)
#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
11.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=janb-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2011')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig("histograma2011.jpg")
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janzz[j] >=0 and janzz[j] <=25:
        count1 = count1+1
print("2011")
print("entre 0 e 25")
print(count1/size)

```

```

count2=0
for j in range (0,size):
    if jan[z[j] >=26 and jan[z[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if jan[z[j] >=51 and jan[z[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if jan[z[j] >=76 and jan[z[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if jan[z[j] >=101 and jan[z[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if jan[z[j] >=126 and jan[z[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if jan[z[j] >=151 and jan[z[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if jan[z[j] >=176 and jan[z[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if jan[z[j] < 0 and jan[z[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

```

```

count10=0
for j in range (0,size):
    if jan[z[j] <=-26 and jan[z[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if jan[z[j] <=-51 and jan[z[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if jan[z[j] <=-76 and jan[z[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if jan[z[j] <=-101 and jan[z[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if jan[z[j] <=-126 and jan[z[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if jan[z[j] <=-151 and jan[z[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if jan[z[j] <=-176 and jan[z[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)
#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
12.txt', delimiter = ',', dtype=float)
df= x.values
k = pd.DataFrame(x)

```

```

new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2012')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2012.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2012")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janb[j] >=26 and janb[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janb[j] >=51 and janb[j] <=75:
        count3 = count3+1

```

```

print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if jan[z[j] >=76 and jan[z[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if jan[z[j] >=101 and jan[z[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if jan[z[j] >=126 and jan[z[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if jan[z[j] >=151 and jan[z[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if jan[z[j] >=176 and jan[z[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if jan[z[j] < 0 and jan[z[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if jan[z[j] <=-26 and jan[z[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if jan[z[j] <=-51 and jan[z[j] >=-75:
        count11 = count11+1

```

```

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if jan[z[j] <=-76 and jan[z[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if jan[z[j] <=-101 and jan[z[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if jan[z[j] <=-126 and jan[z[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if jan[z[j] <=-151 and jan[z[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if jan[z[j] <=-176 and jan[z[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)
#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
13.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.6008333333333
lon=-47.7130555555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]

```

```

So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=janb-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2013')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2013.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janb[j] >=0 and janb[j] <=25:
        count1 = count1+1
print("2013")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janb[j] >=26 and janb[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janb[j] >=51 and janb[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janb[j] >=76 and janb[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):

```



```
    if janz[j] >=101 and janz[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)
```

```
count6=0
for j in range (0,size):
    if janz[j] >=126 and janz[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)
```

```
count7=0
for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1
```

```
print("entre 151 e 175")
print(count7/size)
```

```
count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)
```

```
count9=0
for j in range (0,size):
    if janz[j] < 0 and janz[j] >=-25:
        count9 = count9+1
```

```
print("entre 0 e -25")
print(count9/size)
```

```
count10=0
for j in range (0,size):
    if janz[j] <=-26 and janz[j] >=-50:
        count10 = count10+1
```

```
print("entre -26 e -50")
print(count10/size)
```

```
count11=0
for j in range (0,size):
    if janz[j] <=-51 and janz[j] >=-75:
        count11 = count11+1
```

```
print("entre -51 e -75")
print(count11/size)
```

```
count12=0
for j in range (0,size):
    if janz[j] <=-76 and janz[j] >=-100:
        count12 = count12+1
```

```
print("entre -76 e -100")
print(count12/size)
```

```

count13=0
for j in range (0,size):
    if janz[j] <=-101 and janz[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if janz[j] <=-126 and janz[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if janz[j] <=-151 and janz[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if janz[j] <=-176 and janz[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)
#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
14.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.600833333333
lon=-47.713055555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lonr))/60+(Et/60)
w=(12-hc)*15

```

```

radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')
plt.title('Brasilia 2014')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2014.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janز[j] >=0 and janز[j] <=25:
        count1 = count1+1
print("2014")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janز[j] >=26 and janز[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janز[j] >=51 and janز[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janز[j] >=76 and janز[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if janز[j] >=101 and janز[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if janز[j] >=126 and janز[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

```

```
count7=0
for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1
```

```
print("entre 151 e 175")
print(count7/size)
```

```
count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1
print("entre 176 e 200")
print(count8/size)
```

```
count9=0
for j in range (0,size):
    if janz[j] < 0 and janz[j] >=-25:
        count9 = count9+1
```

```
print("entre 0 e -25")
print(count9/size)
```

```
count10=0
for j in range (0,size):
    if janz[j] <=-26 and janz[j] >=-50:
        count10 = count10+1
```

```
print("entre -26 e -50")
print(count10/size)
```

```
count11=0
for j in range (0,size):
    if janz[j] <=-51 and janz[j] >=-75:
        count11 = count11+1
```

```
print("entre -51 e -75")
print(count11/size)
```

```
count12=0
for j in range (0,size):
    if janz[j] <=-76 and janz[j] >=-100:
        count12 = count12+1
```

```
print("entre -76 e -100")
print(count12/size)
```

```
count13=0
for j in range (0,size):
    if janz[j] <=-101 and janz[j] >=-125:
        count13 = count13+1
```

```
print("entre -101 e -125")
print(count13/size)
```

```
count14=0
for j in range (0,size):
    if janz[j] <=-126 and janz[j] >=-150:
        count14 = count14+1
```

```

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if janz[j] <=-151 and janz[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

count16=0
for j in range (0,size):
    if janz[j] <=-176 and janz[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)
#####
x=pd.read_csv(r'C:\Users\Artur\Desktop\INPE\INPE\2018\arquivos_TXT_dados_prontos\BRB\BRB_20
15.txt', delimiter = ',',dtype=float)
df= x.values
k = pd.DataFrame(x)
new = k.dropna(axis=0)
kkk = new.values
lat=-15.6008333333333
lon=-47.7130555555556
latr=lat*(math.pi/180)
lonr=lon*(math.pi/180)
#CALCULOS
dia=kkk[:,2]
minuto=kkk[:,3]
gl=kkk[:,4]
dire=kkk[:,6]
difu=kkk[:,5]
So=1360
angdia=(2*math.pi)*((dia-1)/365)
Eo=1.00011+0.034221*np.cos(angdia)+0.00128*np.sin(angdia)+0.000719*np.cos(2*angdia)+0.000077*
np.sin(2*angdia)
Sa=Eo*So
Dec=(0.006918-0.399912*np.cos(angdia)+0.070257*np.sin(angdia)-
0.006758*np.cos(2*angdia)+0.000907*np.sin(2*angdia)-
0.002697*np.cos(3*angdia)+0.00148*np.sin(3*angdia))*(180/math.pi)
Et=(0.000075+0.001868*np.cos(angdia)-0.032077*np.sin(angdia)-0.014615*np.cos(2*angdia)-
0.04089*np.sin(2*angdia))*229.18
hc=(minuto/60)-(4*(0-lon))/60+(Et/60)
w=(12-hc)*15
radianodec=(Dec*math.pi)/180
radianolat=(lat*math.pi)/180
radianow=(w*math.pi)/180
cosu=np.sin(radianodec)*np.sin(radianolat)+np.cos(radianodec)*np.cos(radianolat)*np.cos(radianow)
#HISTOGRAMA
jana=kkk[:,4]-kkk[:,5]
janb=kkk[:,6]*cosu
janz=jانب-jana
bins = [-200,-175,-150,-125,-100,-75,-50,-25,0,25,50,75,100,125,150,175,200]
ax=plt.gca()
plt.xlabel('Diferenca')
plt.ylabel('Quantidade')

```

```

plt.title('Brasilia 2015')
plt.hist(janz, bins, histtype = 'bar', rwidth=0.8)
plt.savefig('histograma2015.jpg')
plt.show()
size=len(janz)
count1=0
for j in range (0,size):
    if janz[j] >=0 and janz[j] <=25:
        count1 = count1+1
print("2015")
print("entre 0 e 25")
print(count1/size)

count2=0
for j in range (0,size):
    if janz[j] >=26 and janz[j] <=50:
        count2 = count2+1
print("entre 26 e 50")
print(count2/size)

count3=0
for j in range (0,size):
    if janz[j] >=51 and janz[j] <=75:
        count3 = count3+1
print("entre 51 e 75")
print(count3/size)

count4=0
for j in range (0,size):
    if janz[j] >=76 and janz[j] <=100:
        count4 = count4+1
print("entre 76 e 100")
print(count4/size)

count5=0
for j in range (0,size):
    if janz[j] >=101 and janz[j] <=125:
        count5 = count5+1
print("entre 101 e 125")
print(count5/size)

count6=0
for j in range (0,size):
    if janz[j] >=126 and janz[j] <=150:
        count6 = count6+1
print("entre 126 e 150")
print(count6/size)

count7=0
for j in range (0,size):
    if janz[j] >=151 and janz[j] <=175:
        count7 = count7+1

print("entre 151 e 175")
print(count7/size)

count8=0
for j in range (0,size):
    if janz[j] >=176 and janz[j] <=200:
        count8 = count8+1

```

```

print("entre 176 e 200")
print(count8/size)

count9=0
for j in range (0,size):
    if jan[z[j] < 0 and jan[z[j] >=-25:
        count9 = count9+1

print("entre 0 e -25")
print(count9/size)

count10=0
for j in range (0,size):
    if jan[z[j] <=-26 and jan[z[j] >=-50:
        count10 = count10+1

print("entre -26 e -50")
print(count10/size)

count11=0
for j in range (0,size):
    if jan[z[j] <=-51 and jan[z[j] >=-75:
        count11 = count11+1

print("entre -51 e -75")
print(count11/size)

count12=0
for j in range (0,size):
    if jan[z[j] <=-76 and jan[z[j] >=-100:
        count12 = count12+1

print("entre -76 e -100")
print(count12/size)

count13=0
for j in range (0,size):
    if jan[z[j] <=-101 and jan[z[j] >=-125:
        count13 = count13+1

print("entre -101 e -125")
print(count13/size)

count14=0
for j in range (0,size):
    if jan[z[j] <=-126 and jan[z[j] >=-150:
        count14 = count14+1

print("entre -126 e -150")
print(count14/size)

count15=0
for j in range (0,size):
    if jan[z[j] <=-151 and jan[z[j] >=-175:
        count15 = count15+1

print("entre -151 e -175")
print(count15/size)

```

```
count16=0
for j in range (0,size):
    if janz[j] <=-176 and janz[j] >=-200:
        count16 = count16+1

print("entre -176 e -200")
print(count16/size)
```