



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

sid.inpe.br/mtc-m21b/2016/11.18.17.34-PUD

INTRODUÇÃO AO CLIMATE DATA OPERATORS (VERSÃO ATUALIZADA)

José Guilherme Martins dos Santos

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3MQU4Q8>>

INPE
São José dos Campos
2016

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@inpe.br

COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):

Presidente:

Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação (CPG)

Membros:

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

Dr. André de Castro Milone - Coordenação de Ciências Espaciais e Atmosféricas (CEA)

Dra. Carina de Barros Melo - Coordenação de Laboratórios Associados (CTE)

Dr. Evandro Marconi Rocco - Coordenação de Engenharia e Tecnologia Espacial (ETE)

Dr. Hermann Johann Heinrich Kux - Coordenação de Observação da Terra (OBT)

Dr. Marley Cavalcante de Lima Moscati - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Silvia Castro Marcelino - Serviço de Informação e Documentação (SID) **BIBLIO-**

TECA DIGITAL:

Dr. Gerald Jean Francis Banon

Clayton Martins Pereira - Serviço de Informação e Documentação (SID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Ducca Barbedo - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



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

sid.inpe.br/mtc-m21b/2016/11.18.17.34-PUD

INTRODUÇÃO AO CLIMATE DATA OPERATORS (VERSÃO ATUALIZADA)

José Guilherme Martins dos Santos

URL do documento original:

<<http://urlib.net/8JMKD3MGP3W34P/3MQU4Q8>>

INPE
São José dos Campos
2016



Esta obra foi licenciada sob uma Licença Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada.

This work is licensed under a Creative Commons Attribution-NonCommercial 3.0 Unported License.

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
2 Instalação do CDO com suporte a NetCDF clássico	3
2.1 Instalação da biblioteca NetCDF	3
2.2 Instalação do CDO	5
3 Instalação do CDO via apt-get install sem suporte a NetCDF .	7
4 Instalação do CDO com suporte a NetCDF4 (HDF5) e GRIB2 .	9
4.1 Instalação da biblioteca zlib	10
4.2 Instalação da biblioteca proj.4	10
4.3 Instalação da biblioteca szip	11
4.4 Instalação da biblioteca hdf5	11
4.5 Instalação da biblioteca jasper	11
4.6 Instalação da biblioteca NetCDF4	11
4.7 Instalação da biblioteca grib	12
4.8 Instalação do CDO	12
5 Encadeamento de operadores usando dados NetCDF4 (HDF5) .	15
6 OPERADORES	17
6.1 Visualizar ajuda com o cdo	17
6.2 Encadeamento de operadores	17
6.3 Informações sobre o arquivo	17
6.4 Manipulação de arquivos	21
6.4.1 Operador Copy	21
6.4.2 Operador Cat	21
6.4.3 Operador merge	22
6.4.4 Operador mergetime	22
6.4.5 Operador split	22
6.4.6 Operador splityearmon	23
6.4.7 Operadores de seleção	23
6.4.7.1 Operador select ou delete	23
6.4.7.2 Operador selname	24

6.4.7.3	Operador sellevel	24
6.4.7.4	Operadores selday, selmon e selyear	24
6.4.7.5	Operador sellonlatbox	25
6.4.7.6	Operador selseason	25
6.5	Operadores de comparação	25
6.6	Operadores de modificação de metadados e arquivos	26
6.6.1	Operador settaxis	26
6.6.2	Operador setcalendar	27
6.6.3	Operador chname	27
6.6.4	Operador inverlat	27
6.6.5	Operador para valores ausentes ou indefinidos	27
6.7	Operadores aritméticos	28
6.7.1	Operadores matemáticos	28
6.7.2	Operações com constantes	29
6.7.3	Operações usando dois conjunto de dados	30
6.8	Cálculos estatísticos	30
6.8.1	Média de vários arquivos (ensemble)	31
6.8.2	Campos bidimensionais	31
6.8.3	Cálculo estatístico zonal	31
6.8.4	Cálculo estatístico meridional	32
6.8.5	Cálculo estatístico vertical	32
6.8.6	Cálculo estatístico temporal	33
6.8.7	Cálculo estatístico com média móvel	33
6.8.8	Cálculo estatístico sobre todos os tempos	34
6.8.9	Cálculo estatístico diário	34
6.8.10	Cálculo estatístico mensal	35
6.8.11	Cálculo estatístico anual	35
6.8.12	Cálculo estatístico sazonal	36
6.8.13	Valor estatístico mensal de vários anos	37
6.8.14	Valor estatístico sazonal de vários anos	37
6.9	Interpolação	38
6.9.1	Operador remapbil	38
6.10	Importação e exportação	38
6.10.1	Importação de conjunto de dados binários	38
6.10.2	Conversão de arquivo texto para NetCDF	38
6.10.3	Extrair arquivos ASCII de NetCDF	39
6.10.4	Correlação	39

6.10.5	Correlação espacial	39
6.10.6	Correlação temporal	40
7	MÓDULO PRÁTICO	41
7.1	Alterando a coordenada vertical	41
7.2	Alterando os valores NaN do arquivo NetCDF	42
7.3	Alterando valores do arquivo NetCDF	43
7.4	Anomalia climatológica zonal de altura geopotencial	44
7.5	Calculando a relação entre os desvios padrão simulado e observado	45
7.6	Calculando a diferença entre o tempo posterior e o anterior	45
7.7	Calculando a anomalia padronizada	46
7.8	Calculando a velocidade do vento	46
7.8.1	Método 1	46
7.8.2	Método 2	46
7.9	Calculando o erro absoluto médio (EAM)	47
7.10	Calculando o número de dias consecutivos de precipitação	47
7.11	Calculando o bias	48
7.12	Calculando média espacial sobre a Bacia Amazônica	48
7.13	Calculando RMSE	49
7.14	Calculando “pentadas”	50
7.15	Convertendo um arquivo de perfil vertical formato texto para NetCDF	52
7.16	Criando anomalias	53
7.17	Criando climatologia	53
7.18	Criando máscara para oceano ou continente	53
7.19	Extraindo a série temporal de um ponto com diversas informações	54
7.20	Extraindo apenas a série temporal de um ponto	54
7.21	Interpolando mapa de vegetação	54
7.22	Mascarando regiões	55
7.23	Mascarando valores de velocidade e vetor do vento	56
7.24	Preenchimento de dados ausentes	57
7.25	Redução de dimensão de arquivos netCDF	57
8	Links interessantes	59
	REFERÊNCIAS BIBLIOGRÁFICAS	61

1 INTRODUÇÃO

O Climate Data Operators (CDO) representa um conjunto de comandos estatísticos e aritméticos úteis para processar dados meteorológicos no formato GRIB e NetCDF. A facilidade em usar essa ferramenta está no fato de que os comandos são executados diretamente no terminal do Linux. O usuário pode escrever um script utilizando a linguagem mais apropriada para automatizar suas tarefas. O CDO não possui ambiente gráfico, ele apenas processa dados e a visualização é feita com o programa de sua preferência.

O CDO é uma alternativa para processar dados por meio dos seus operadores. Essa tarefa poderia ser feita por uma linguagem de programação, como por exemplo, o fortran, mas para quem não está habituado a programar, esse software é capaz de realizar tarefas bem robustas, como interpolação, cálculo de médias, calculos estatísticos, dentre outros.

O CDO possui uma lista de discussão onde usuários estão sempre disponíveis a ajudar. O endereço eletrônico da lista de discussão está disponível em:

<https://code.zmaw.de/projects/cdo/boards/2>

Algumas características do CDO são:

- Existem mais de 600 operadores que podem ser empregados para manipulação de arquivos;
- Interface amigável para usuários Linux;
- Os dados podem ser processados por mais de um operador;
- Suporte a diferentes tipos de grades (regular, não-estruturada, gaussiana e curvilinear);
- Compatível com sistemas UNIX/Linux, Cygwin e MacOS;
- Fórum para tirar dúvidas.

IPC: Nativamente, o CDO tem apenas suporte para dados grib. Para usá-lo com arquivos NetCDF é recomendável a instalação da biblioteca NetCDF. Os passos a seguir mostrarão como instalar essa biblioteca.

2 Instalação do CDO com suporte a NetCDF clássico

2.1 Instalação da biblioteca NetCDF

A biblioteca NetCDF e o CDO foram instalados no Linux Ubuntu 32 bits.

A biblioteca NetCDF a ser instalada será a versão 4.3.3.1 sem suporte a NetCDF4 (HDF5) e GRIB2.

Para fazer o download do arquivo, acesse o endereço eletrônico:

<http://www.unidata.ucar.edu/downloads/netcdf/index.jsp>

E faça o download da versão mais recente, neste tutorial é utilizada a versão 4.3.3.1 (Figura 7.2). Ao clicar em *The Latest Stable netCDF-C Release, tar.gz form* (*The netCDF-C 4.3.3.1 release is the latest stable release, netcdf-4.3.3.1.tar.gz.*) o arquivo será salvo no diretório Downloads do seu Linux.

NetCDF Downloads

NetCDF (network Common Data Form) is a set of software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data. Distributions are provided for Java and C/C++/Fortran. See the [netCDF web site](#) and the FAQ answer to [How do I get the netCDF software package?](#) for more information.

NetCDF-Java Stable Releases/releases

Downloads of stable releases of netCDF-Java.

The netCDF-Java library, version 4.

The NetCDF-Java Library is a Java interface to "Common Data Model" files, using the netCDF API.

The netCDF-Java library, version 2.2.

Older, more limited version of the NetCDF-Java Library.

NetCDF-C Releases

Downloads of stable and beta releases of the netCDF C Library.

The Latest Stable netCDF-C Release, tar.gz form

The netCDF-C 4.3.3.1 release is the latest stable release, netcdf-4.3.3.1.tar.gz.

The Latest Stable netCDF-C Release, .zip form

The netCDF-C 4.3.3.1 release is the latest stable release, netcdf-4.3.3.1.zip.

Pre-built Windows Binaries for the latest version of NetCDF-C

We currently provide binary distributions for Windows, only. More information can be found at the link above.

All netCDF-C library and utilities source code releases.

All current and historic releases of the netCDF C library source code, including pre-release code.

The netCDF-C development source repository

The GitHub repository for the netCDF C source code. Unless you are interested in working with the development version of netCDF-C, you will want to use one of the releases linked above. If you choose work with the development branch, you will need to generate the 'configure' script using 'autoreconf -i -f'.

Figura 2.1 - Download da biblioteca NetCDF.

Para descompactar o arquivo **netcdf-4.3.3.1.tar.gz**, digite:

```
tar -zxvf netcdf-4.3.3.1.tar.gz
```

Será criado o diretório **netcdf-4.3.3.1**, entre nele e digite o comando abaixo.

Não copie e cole os comandos no seu terminal Linux porque isso vai gerar erro. Digite os comandos.

```
sudo ./configure --disable-netcdf-4
```

Esse procedimento irá instalar as bibliotecas em **/usr/local/lib** e os executáveis em **/usr/local/bin** que é a instalação padrão.

Caso apareça o erro abaixo após digitar o comando acima:

```
configure: error: Cannot find m4 utility. Install m4 and try again.
```

Digite o comando abaixo para instalar o m4:

```
sudo apt-get install m4
```

E depois:

```
sudo make && sudo make install
```

Aparecerá uma mensagem dizendo que o NetCDF foi instalado corretamente.

Digite no seu terminal Linux o comando **ncdump**, caso apareça o erro abaixo, proceda da seguinte forma:

```
gui@zeus: /Downloads/netcdf-4.2.1.1$ ncdump ncdump: error while loading shared libraries: libnetcdf.so.7: cannot open shared object file: No such file or directory
```

Vá para o diretório **HOME** e adicione a linha abaixo no seu **.bashrc**. Para ver esse arquivo, digite **ls -a**, a opção **-a** é para visualizar arquivos ocultos do sistema no diretório corrente.

```
export LD_LIBRARY_PATH=/usr/local/lib:/usr/lib:/lib
```

Não esqueça de atualizar o seu **.bashrc** digitando no seu **HOME**:

```
source .bashrc
```

Em seguida, digite novamente **ncdump**. Se o erro desapareceu significa que o NetCDF foi instalado corretamente.

Dessa forma, a futura instalação do CDO terá suporte para dados NetCDF. Caso o usuário não faça isso, o CDO terá apenas suporte a dados GRIB1.

2.2 Instalação do CDO

Para instalar o CDO com suporte a biblioteca NetCDF instalada no passo anterior, acesse o site:

<https://code.zmaw.de/projects/cdo/files>

e faça o download da última versão. Há versões para Windows e Linux.

E para quem deseja compilar o código (recomendável) usando o Ubuntu, realizar o download no site acima da última versão do CDO. Este material utiliza a versão 1.7.2.

Ao clicar no arquivo **cdo-1.7.2.tar.gz**, ele será salvo no diretório Downloads. Vá para esse diretório e descompacte esse arquivo usando o comando

```
tar -zxvf cdo-1.7.2.tar.gz
```

Será criado o diretório **cdo-1.7.2**. Entre nesse diretório, e digite:

```
sudo ./configure --with-netcdf=/usr/local
```

E depois:

```
sudo make && sudo make install
```

Essa opção por padrão instalará o executável do CDO em: **/usr/local/bin**

Com esses passos, o CDO terá suporte a dados NetCDF e GRIB.

3 Instalação do CDO via apt-get install sem suporte a NetCDF

Para usuários Ubuntu, basta digitar no seu terminal:

```
sudo apt-get install cdo
```

Será solicitada sua senha, apenas digite-a.

Lembrando que a versão instalada por esse processo não é a versão mais nova do CDO, porém serve para processar seus dados apenas no formato GRIB1.

Outra possibilidade de instalação do CDO com suporte a dados NetCDF4 (HDF5) e GRIB2 é apresentada abaixo. Essa instalação é opcional.

4 Instalação do CDO com suporte a NetCDF4 (HDF5) e GRIB2

IPC: Essa instalação é opcional, caso o usuário apenas utilize NetCDF clássico e GRIB1 não há necessidade de instalar essas bibliotecas. Utilize apenas o passo 2.

Alguns centros meteorológicos estão disponibilizando dados no formato NetCDF4 (HDF5) ou GRIB2 e o CDO tem suporte para esses formatos. Os procedimentos a seguir mostrarão como instalar as bibliotecas necessárias para processar esses formatos.

Por padrão, todas as bibliotecas instaladas ficarão em: `/usr/local/lib` e os executáveis em `/usr/local/bin`.

Download das bibliotecas a serem instaladas. Utilize sempre a versão mais recente:

- CDO: <https://code.zmaw.de/projects/cdo/files>
- NETCDF: <http://www.unidata.ucar.edu/downloads/netcdf/index.jsp>
- GRIB: <https://software.ecmwf.int/wiki/display/GRIB/Releases>
- JASPER: <http://www.ece.uvic.ca/~frodo/jasper/#download>
- HDF5: <ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-4>
- SZIP: <ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-4>
- ZLIB: <ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-4>
- PROJ.4: <https://trac.osgeo.org/proj/>

Ao realizar o download dos arquivos acima, descompacte-os.

```
tar -zxvf cdo-1.6.8.tar.gz
tar -zxvf netcdf-4.3.3.1.tar.gz
tar -zxvf grib_api-1.13.0.tar.gz
unzip jasper-1.900.1.zip
tar -zxvf hdf5-1.8.13.tar.gz
tar -zxvf szip-2.1.tar.gz
tar -zxvf zlib-1.2.8.tar.gz
unzip proj.4-master.zip
```

Ou clique com o botão direito do mouse sobre o arquivo e selecione a opção *Extrair aqui*.

IPC: NÃO COPIE E COLE OS COMANDOS ABAIXO, APENAS DIGITE-OS NO TERMINAL DO LINUX.

Antes de iniciar a instalação certifique-se de digitar os comandos abaixo.

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install csh
sudo apt-get install g++
sudo apt-get install gfortran
sudo apt-get install m4
sudo apt-get install automake
sudo apt-get install libtool
```

4.1 Instalação da biblioteca zlib

Entre no diretório zlib-1.2.8 e digite no terminal do Linux:

```
sudo ./configure --prefix=/usr/local
```

E depois,

```
sudo make && sudo make install
```

4.2 Instalação da biblioteca proj.4

Entre no diretório proj.4-master e digite no terminal do Linux:

```
./autogen.sh
```

O comando acima criará o arquivo configure. A seguir, digite:

```
sudo ./configure --prefix=/usr/local
```

E depois,

```
sudo make && sudo make install
```

4.3 Instalação da biblioteca szip

Entre no diretório szip-2.1 e digite no terminal do Linux:

```
sudo ./configure --prefix=/usr/local
```

E depois,

```
sudo make && sudo make install
```

4.4 Instalação da biblioteca hdf5

Entre no diretório hdf5-1.8.13 e digite no terminal do Linux:

```
sudo ./configure --with-zlib=/usr/local --with-szlib=/usr/local  
--prefix=/usr/local --enable-shared --enable-fortran --enable-cxx
```

E depois,

```
sudo make && sudo make install
```

4.5 Instalação da biblioteca jasper

Entre no diretório jasper-1.900.1 e digite no terminal do Linux:

```
sudo ./configure --with-pic --prefix=/usr/local
```

E depois,

```
sudo make && sudo make install
```

4.6 Instalação da biblioteca NetCDF4

Entre no diretório netcdf-4.3.3.1 e digite as linhas abaixo no terminal do Linux:

```
export CPPFLAGS=-I/usr/local/include  
export LDFLAGS=-L/usr/local/lib  
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
```

```
sudo ./configure --enable-netcdf4 --enable-shared  
--prefix=/usr/local
```

E depois,

```
sudo make && sudo make install
```

4.7 Instalação da biblioteca grib

Entre no diretório `grib_api-1.13.0` e digite no terminal do Linux:

```
export CPPFLAGS=-I/usr/local/include
export LDFLAGS=-L/usr/local/lib
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
export CFLAG="fPIC"
export CC=gcc
export FC=gfortran
```

```
sudo ./configure --prefix=/usr/local --with-netcdf=/usr/local
--with-jasper=/usr/local
```

E depois,

```
sudo make && sudo make install
```

4.8 Instalação do CDO

Entre no diretório `cdo-1.7.2` e digite no terminal do Linux:

```
sudo ./configure --with-netcdf=/usr/local --with-jasper=/usr/local
--with-hdf5=/usr/local --with-grib_api=/usr/local --with-
proj=/usr/local --prefix=/usr/local
```

E depois,

```
sudo make && sudo make install
```

Assim que tudo estiver instalado, não esqueça de adicionar no seu `.bashrc` a linha abaixo:

```
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH:/usr/lib:/lib
```

Não esqueça de atualizar o seu `.bashrc` digitando no seu diretório `HOME`:

```
source .bashrc
```

Após o término da instalação, digite no terminal do Linux `cdo -V`:

O que está destacado em vermelho são as novas funcionalidade instaladas, isto é, CDO com suporte a NetCDF4 e GRIB2.

```
Climate Data Operators version 1.7.2 (http://mpimet.mpg.de/cdo)
Compiled: by root on curupira (x86_64-unknown-linux-gnu) Dec 10 2015 14:31:03
Compiler: gcc -std=gnu99 -g -O2 -fopenmp
version: gcc (Ubuntu 4.8.4-2ubuntu1 14.04) 4.8.4
Features: DATA PTHREADS OpenMP HDF5 NC4/HDF5 Z PROJ.4 SSE2
Libraries: HDF5/1.8.13 proj/4.92 Filetypes: srv ext ieg grb grb2 nc nc2 nc4 nc4c
CDI library version : 1.7.0 of Dec 10 2015 14:30:13
CGRIBEX library version : 1.7.3 of Sep 14 2015 10:58:44
GRIB_API library version : 1.13.0
netCDF library version : 4.3.3.1 of Nov 30 2015 09:58:54 $
HDF5 library version : 1.8.13
SERVICE library version : 1.4.0 of Dec 10 2015 14:29:58
EXTRA library version : 1.4.0 of Dec 10 2015 14:29:53
IEG library version : 1.4.0 of Dec 10 2015 14:29:56
FILE library version : 1.8.2 of Dec 10 2015 14:29:53
```

Feito todos esses passos, o CDO terá suporte a NetCDF4 (HDF5) e GRIB2.

Como saber o formato do arquivo? Digite no terminal Linux o comando **cdo show-format nome_do_arquivo.nc**. O resultado será o tipo de arquivo, isto é, dependerá da extensão a ser utilizada.

Outra possibilidade é digitar **ncdump -k nome_do_arquivo.nc**.

A seguir, serão apresentados alguns tópicos sobre a utilização dos operadores. Para informações adicionais, leia o manual do CDO disponível em:

<https://code.zmaw.de/projects/cdo/wiki/CdoDocumentation>

5 Encadeamento de operadores usando dados NetCDF4 (HDF5)

Ao tentar utilizar vários operadores com dados NetCDF4, alguns erros podem surgir, por exemplo:

```
*** Error in 'cdo': malloc(): memory corruption (fast): 0x00007f857007d7c0 ***  
Aborted
```

Ou:

```
Falha de segmentação (imagem do núcleo gravada)
```

Outro tipo de erro que é o mais comum:

```
Error (xxx) : NetCDF: HDF error  
cdo(xxx) malloc: *** error for object xxx: pointer being freed was not allocated  
segmentation fault (core dumped)  
Bus error (core dumped)
```

Isso ocorre porque a biblioteca hdf5 não foi compilada com a opção “`--enable-threadsafe`”.

Como saber o formato do dado? Basta utilizar o operador `showformat`:

```
Exemplo: cdo showformat prec.GPCP.jan1979.nc  
NetCDF4 classic ZIP
```

Esse dado é do tipo NetCDF4 classic, e por isso, os erros porque a biblioteca hdf5 não foi compilada com a opção “`--enable-threadsafe`”.

Para resolver esse problema há duas soluções:

Solução1: Mudar de NetCDF4 classic para NetCDF padrão por meio da opção “`-f nc`”.

```
cdo -f nc -fldmean -sellonlatbox,-90,-30,-60,10 prec.GPCP.jan1979.nc ppt.SA.nc
```

Solução2: Deixar o comando serial com a opção “`-L`” e o dado continuará no formato NetCDF4 classic.

```
cdo -L -fldmean -sellonlatbox,-90,-30,-60,10 prec.GPCP.jan1979.nc ppt.SA.nc
```

Essa solução está disponível em:

<https://code.zmaw.de/projects/cdo/wiki#Errors-with-operator-chaining-and-netCDF4HDF5-files>

6 OPERADORES

Convenção utilizada nesse tutorial:

if.nc representa o dado de entrada e **of.nc** é o arquivo de saída.

6.1 Visualizar ajuda com o cdo

Quando necessário o usuário pode digitar o comando `--help` ou `-h` para obter informações sobre algum operador.

Por exemplo:

```
cdo --help fillmiss
```

ou

```
cdo -h fillmiss
```

6.2 Encadeamento de operadores

Entende-se por encadeamento de operadores a utilização de vários operadores ao mesmo tempo. Isso reduz o espaço em disco e agiliza o processamento dos dados. Para utilizar essa funcionalidade deve-se utilizar o símbolo “-” antes de cada operador que somente funciona quando ele tem um número fixo de entrada e saída.

Por exemplo:

```
cdo -selmon,1 -sellevel,1000 -sellonlatbox,-100,-20,-60,20 if.nc of.nc
```

O uso do encadeamento somente foi possível com o uso do símbolo “-”. Os comandos são executados sempre da direita para a esquerda. Nesse caso o primeiro operador realiza um recorte do dado no domínio selecionado, em seguida seleciona o nível vertical de 1000hPa e por fim, seleciona o mês 1, ou seja, janeiro.

6.3 Informações sobre o arquivo

Como exemplo, será utilizado o arquivo mensal de precipitação do GPCP para o ano de 2012 (gpcp.2012) e o de temperatura do ar (temp.ar.2010.nc) em vários níveis verticais do NCEP .

- **infon:** informações sobre o conjunto de dados listado pelo nome da variá-

vel.

Exemplo: Digite o comando `cdo infon gpcp.2012.nc` que aparecerão as informações abaixo:

```
-1 :      Date      Time  Level Gridsize  Miss :      Minimum      Mean      Maximum : Parameter name
 1 : 2012-01-01 00:00:00    0   10368    0 :      0.0000      2.1176    20.782 : precip
 2 : 2012-02-01 00:00:00    0   10368    0 :      0.0000      2.1220    18.419 : precip
 3 : 2012-03-01 00:00:00    0   10368    0 :      0.0000      2.1849    20.296 : precip
 4 : 2012-04-01 00:00:00    0   10368    0 :      0.0000      2.1045    16.895 : precip
 5 : 2012-05-01 00:00:00    0   10368    0 :      0.0000      2.1011    17.928 : precip
 6 : 2012-06-01 00:00:00    0   10368    0 :      0.0000      2.2335    25.607 : precip
 7 : 2012-07-01 00:00:00    0   10368    0 :      0.0000      2.2797    20.883 : precip
 8 : 2012-08-01 00:00:00    0   10368    0 :      0.0000      2.3429    26.608 : precip
 9 : 2012-09-01 00:00:00    0   10368    0 :      0.0000      2.3149    19.403 : precip
10 : 2012-10-01 00:00:00    0   10368    0 :      0.0000      2.1460    17.941 : precip
11 : 2012-11-01 00:00:00    0   10368    0 : 1.1622e-05      2.0858    20.259 : precip
12 : 2012-12-01 00:00:00    0   10368    0 :      0.0000      2.1242    19.204 : precip
cdo infon: Processed 124416 values from 1 variable over 12 timesteps ( 0.00s )
```

Ao digitar esse comando serão geradas nove colunas na seguinte ordem:

- data (Date): data da variável.
- tempo (Time): hora da variável.
- nível (Level): nível da variável. É zero porque não há níveis verticais.
- tamanho do domínio (Gridsize): número de pontos em x vezes o número de pontos em y.
- valores ausentes (Miss): se há valores ausentes no seu dado. Zero quer dizer que não há dado ausente.
- valor mínimo (Minimum): valor mínimo para cada tempo do arquivo. Não diz qual é a latitude/longitude onde ocorre o valor mínimo.
- valor médio (Mean): valor médio para cada tempo do arquivo. Não diz qual é a latitude/longitude onde ocorre o valor médio.
- valor máximo (Maximum): valor máximo para cada tempo do arquivo. Não diz qual é a latitude/longitude onde ocorre o valor máximo.
- nome da variável (Parameter name): nome da variável do arquivo.
- **sinfon:** informações sobre o conjunto de dados, porém de forma reduzida.

Exemplo: `cdo sinfon gpcp.2012.nc`

```

File format : netCDF
-1 : Institut Source  Ttype  Levels Num  Points Num Dtype : Parameter name
 1 : unknown GPCP    instant    1  1    10368  1  F32  : precip
Grid coordinates :
 1 : lonlat          : points=10368 (144x72)
                        lon : 1.25 to 358.75 by 2.5 degrees_east  circular
                        lat : 88.75 to -88.75 by -2.5 degrees_north

Vertical coordinates :
 1 : surface         : levels=1
Time coordinate : 12 steps
  RefTime = 1800-01-01 00:00:00 Units = days  Calendar = standard
YYYY-MM-DD hh:mm:ss  YYYY-MM-DD hh:mm:ss  YYYY-MM-DD hh:mm:ss  YYYY-MM-DD hh:mm:ss
2012-01-01 00:00:00  2012-02-01 00:00:00  2012-03-01 00:00:00  2012-04-01 00:00:00
2012-05-01 00:00:00  2012-06-01 00:00:00  2012-07-01 00:00:00  2012-08-01 00:00:00
2012-09-01 00:00:00  2012-10-01 00:00:00  2012-11-01 00:00:00  2012-12-01 00:00:00
cdo sinfon: Processed 1 variable over 12 timesteps ( 0.00s )

```

- **pardes:** descrição das variáveis do arquivo

Exemplo: cdo pardes gpcp.2012.nc

- **nlevel:** número de níveis verticais

Exemplo: cdo nlevel temp.ar.2010.nc

- **nyear:** número de anos

Exemplo: cdo nyear gpcp.2012.nc

- **nmon:** número de meses

Exemplo: cdo nmon gpcp.2012.nc

- **ndate:** número de dias

Exemplo: cdo ndate gpcp.2012.nc

- **ntime:** número de *timestep*

Exemplo: cdo ntime gpcp.2012.nc

- **showformat:** mostra o formato do dado

Exemplo: cdo showformat temp.ar.2010.nc

- **showname:** mostra o nome da variável

Exemplo: cdo showname gpcp.2012.nc

- **showlevel:** mostra os níveis verticais

Exemplo: cdo showlevel temp.ar.2010.nc

- **showyear:** mostra os anos

Exemplo: cdo showyear gpcp.2012.nc

- **griddes**: mostra informações sobre o domínio

Exemplo: `cdo griddes gpcp.2012.nc`

```
#
# gridID 1
#
gridtype = lonlat
gridsize = 10368
xname    = lon
xlongname = Longitude
xunits   = degrees_east
yname    = lat
ylongname = Latitude
yunits   = degrees_north
xsize    = 144
ysize    = 72
xfirst   = 1.25
xinc     = 2.5
yfirst   = 88.75
yinc     = -2.5
cdo griddes: Processed 1 variable ( 0.00s )
```

- **zaxisdes**: mostra informações sobre a coordenada vertical

Exemplo: `cdo zaxisdes temp.ar.2010.nc`

```
#
# zaxisID 1
#
zaxistype = pressure
size      = 17
name      = level
longname  = Level
units     = millibar
levels    = 1000 925 850 700 600 500 400 300 250 200 150 100 70 50 30 20 10
cdo zaxisdes: Processed 1 variable ( 0.00s )
```

Há outros comandos para obter informações sobre o arquivo. Não deixe de ler o tutorial que está disponível em:

<https://code.zmaw.de/projects/cdo/wiki/CdoDocumentation>

6.4 Manipulação de arquivos

Importante: Ao processar os dados, eles não serão deletados e nem modificados, apenas serão criados novos arquivos.

6.4.1 Operador Copy

Muda a extensão ou duplica arquivos.

Exemplo: Transformar de NetCDF para grib:

```
cdo -f grb copy gpcp.2012.nc gpcp.2012.grb
```

Em que: **gpcp.2012.nc** pode ser qualquer arquivo NetCDF e **gpcp.2012.grb** é o nome que o usuário escolhe para o arquivo de saída. O arquivo NetCDF não será apagado e nem alterado, apenas será criado um novo arquivo no formato grib.

-f grb: converte para grib

Para converter em NetCDF:

```
cdo -f nc copy if.grb of.nc
```

Exemplo: Juntar arquivos com diferentes tempos: Supondo que você tenha vários arquivos de uma mesma variável, por exemplo, temperatura em três arquivos com as mesmas dimensões, porém com número de tempos distintos. O arquivo1 apresenta 10 tempos, o arquivo2 20 tempos e o arquivo3 5 tempos, para juntar todos em um único arquivo basta fazer:

```
cdo copy y1980.nc y1981.nc y1982.nc temp.1980.1982.nc
```

y1980.nc y1981.nc y1982.nc são os seus arquivos.

temp.1980.1982.nc é o arquivo que será gerado.

6.4.2 Operador Cat

O cat serve também para juntar arquivos. A única diferença está no fato que se o usuário repetir o mesmo comando, as novas informações serão adicionadas no final do arquivo criando uma duplicidade de informações, por isso, tenha cuidado ao usar o cat. Dê preferência pelo copy para juntar arquivos.

Exemplo: `cdo cat y1980.nc y1981.nc y1982.nc temp.1980.1982.nc`

Caso o usuário repita o comando acima, as informações dos arquivos y1980.nc, y1981.nc e y1982.nc serão adicionadas no fim do arquivo temp.1980.1982.nc criando a duplicidade de informações.

6.4.3 Operador merge

Junta arquivos com diferentes variáveis. Lembrando que os arquivos devem possuir as mesmas dimensões.

Exemplo: Juntar os arquivos de temperatura e altura geopotencial.

Exemplo: **cdo merge tar.nc alt.geo.nc tar.alt.geo.nc**

tar.nc e **alt.geo.nc** são os arquivos de entrada e **tar.alt.geo.nc** é o arquivo de saída que conterà as duas variáveis.

6.4.4 Operador mergetime

Junta arquivos de mesma variável (aumenta o número de tempos do arquivo), por exemplo, a temperatura apresenta vários arquivos separados por tempo e o objetivo será unir todos em um arquivo.

Exemplo: Juntar os arquivos de temperatura.

Exemplo: **cdo mergetime tar.1980.nc tar.1981.nc tar.1982.nc tar.1980.1982.nc**

Os arquivos de entrada são tar.1980.nc, tar.1981.nc e tar.1982.nc e o resultado será armazenado em tar.1980.1982.nc.

6.4.5 Operador split

Separa o arquivo em horas, em dias, em meses ou em anos, isso dependerá do operador a ser empregado.

Operador: splitlevel, splithour, splitday, splitmon, splitseas e splityear

Exemplo: Separar o arquivo em meses:

cdo splitmon tar.nc mes.

Onde: **mes.** representa um prefixo qualquer e **tar.nc** é o arquivo de entrada.

Com isso, serão criados 12 arquivos do tipo: mes.01.nc, mes.02.nc, ..., mes.12.nc. Por exemplo, o arquivo mes.01.nc contém todos os janeiros do seu arquivo, o mesmo raciocínio é válido para os demais arquivos.

6.4.6 Operador splityearmon

Separa o arquivo em anos e meses. O arquivo se chama gpcp.as.2000.2001.nc que é um dado de precipitação mensal sobre a América do Sul para os anos de 2000 e 2001.

Exemplo: cdo splityearmon gpcp.as.2000.2001.nc prec.

Em que **prec.** é um apenas um prefixo escolhido pelo usuário, ele servirá para nomear os arquivos que serão gerados.

Serão gerados vários arquivos com os seguintes nomes:

prec.200001.nc, prec.200002.nc, prec.200003.nc, prec.200004.nc, prec.200005.nc,
prec.200006.nc, prec.200007.nc, prec.200008.nc, prec.200009.nc, prec.200010.nc,
prec.200011.nc, prec.200012.nc, prec.200101.nc, prec.200102.nc, prec.200103.nc,
prec.200104.nc, prec.200105.nc, prec.200106.nc, prec.200107.nc, prec.200108.nc,
prec.200109.nc, prec.200110.nc, prec.200111.nc e prec.200112.nc

Observe que são dois anos (2000 e 2001) separados por meses (01, 02, 03, ..., 12).

6.4.7 Operadores de seleção

6.4.7.1 Operador select ou delete

Seleciona ou deleta campos do arquivo de entrada. O arquivo se chama ur.tar.as.2008.2010.nc e nele há duas variáveis, isto é, umidade relativa (rhum) e temperatura do ar (air) mensal em vários níveis verticais para os anos de 2008 a 2010.

Será selecionada apenas a variável **air** para o nível de 1000hpa para os meses de junho (6), julho (7) e agosto (8).

Exemplo:

```
cdo select,name=air,level=1000,month=6,7,8 ur.tar.as.2008.2010.nc  
air.1000hpa.jja.nc
```

Será gerado o arquivo `air.1000hpa.jja.nc` com todos os meses 6,7,8 para 2008, 6,7,8 para 2009 e 6,7,8 para 2010 apenas para o nível de 1000hpa.

Em vez de utilizar 6,7,8 há possibilidade de utilizar a “/” para expandir os números, por exemplo, 6/7 é o mesmo que 6,7,8. Outro exemplo, 2/6 é o mesmo que 2,3,4,5,6,7,8.

Os campos que podem ser selecionados com este operador são: **name** (string), **param** (string), **code** (integer ou inteiro), **ltype** (integer ou inteiro), **levidx** (integer ou inteiro), **level** (float ou real), **minute** (integer ou inteiro), **hour** (integer ou inteiro), **day** (integer ou inteiro), **month** (integer ou inteiro), **year** (integer ou inteiro), **timestep**, (integer ou inteiro) e **timestep_of_year** (integer ou inteiro).

Para deletar informações utilize o operador delete.

6.4.7.2 Operador selname

Seleciona variáveis dentro do arquivo e os salva em um novo arquivo.

Exemplo: `cdo selname,air ur.tar.as.2008.2010.nc temp.nc`

air é o nome da variável que existe dentro de `ur.tar.as.2008.2010.nc` e **temp.nc** é o arquivo de saída que contém o nome da variável **air**.

6.4.7.3 Operador sellevel

O mesmo pode ser feito para selecionar um ou mais níveis verticais de um arquivo, para isso, usa-se o **sellevel**. No exemplo abaixo, foi extraído o nível de 200hPa do **if.nc**, e posteriormente foi criado o novo arquivo **of.nc**.

Exemplo: `cdo sellevel,200 temp.ar.2010.nc temp.200.nc`

6.4.7.4 Operadores selday, selmon e selyear

Para selecionar tempos específicos, utiliza-se o **selday**, **selmon** e **selyear**. Há outros operadores disponíveis e o usuário deve selecionar o melhor para sua aplicação.

Usando o operador **selday** para seleciona dias específicos dado uma lista de dias.

Exemplo: `cdo selday,1,4,7 if.nc of.nc`

Com o comando acima, foram selecionados os dias 1, 4 e 7. A vírgula separa a lista

de dias que deve ser um valor inteiro.

Para seleccionar os meses deve-se utilizar o mesmo raciocínio, isto é:

Exemplo: **cdo selmon,2,4 if.nc of.nc**

Foram seleccionados apenas os meses 2 e 4 e esses meses foram salvos no arquivo de saída **of.nc**.

6.4.7.5 Operador sellonlatbox

Recorta o dado em uma área seleccionada.

Exemplo: **cdo sellonlatbox,-90,-30,-50,10 if.nc of.nc**

A convecção será sempre: longitude oeste, longitude leste, latitude sul e latitude norte.

6.4.7.6 Operador selseason

Selecciona uma dada estação do ano no formato string (texto). É possível seleccionar quantos meses forem necessários, e isso dependerá do usuário. O formato válido é JFMAMJJASOND.

Exemplo1: **cdo selseason,DJFMA if.nc of.nc**

Exemplo2: **cdo selseason,JJAS if.nc of.nc**

6.5 Operadores de comparação

Esses operadores quando utilizados atribuem o valor 1 (um) quando a condição for verdadeira e 0 (zero) quando for falsa.

- eq \Rightarrow igual
- ne \Rightarrow diferente
- le \Rightarrow menor igual
- lt \Rightarrow menor que
- ge \Rightarrow maior igual
- gt \Rightarrow maior que

Exemplo: `cdo eq if1.nc if2.nc of.nc`

Quando a variável do `if1.nc` for igual a variável do `if2.nc` receberá o valor 1, caso contrário, receberá o valor 0 e o resultado será gravado no `of.nc`.

Caso o usuário queira comparar a sua variável com uma constante, utilizam-se os operadores abaixo, a diferença em relação ao operador anterior é o acréscimo da letra **c** ao operador.

- `eqc` \Rightarrow igual a contante
- `nec` \Rightarrow diferente da constante
- `lec` \Rightarrow menor igual a constante
- `ltc` \Rightarrow menor que a constante
- `gec` \Rightarrow maior igual a constante
- `gtc` \Rightarrow maior que a constante

Exemplo: `cdo eqc,2 if.nc of.nc`

Os pontos de grade onde `if.nc` é igual a 2, receberá 1, e no caso contrário, 0.

6.6 Operadores de modificação de metadados e arquivos

6.6.1 Operador `settaxis`

Fixa uma data de referência para a dimensão tempo.

Exemplo: `cdo -r settaxis,2000-01-01,00:00:00,1mon if.nc of.nc`

-r = adiciona um eixo de tempo relativo. **2000-01-01** = defina conforme o seu interesse. **00:00:00** = uma hora qualquer, poderia ser 06:00:00, 12:00:00 ou 18:00:00. **1mon** = intervalo de tempo do arquivo (dt). Algumas possibilidades de incremento: `hour`, `day`, `mon` e `year`

IPC: Sabe aquele arquivo que você tenta abrir no GrADS que mostra o seguinte erro?

```
ga-> sdfopen output.nc Scanning self-describing file: output.nc SDF file
has no discernable time coordinate – using default values. gadsdf: SDF
file does not have any non-coordinate variables.
```

Pois é, seus problemas acabaram! Basta usar esse comando para fixar um eixo de tempo. O GrADS está dizendo que o seu dado não possui a dimensão tempo.

6.6.2 Operador setcalendar

Este operador é útil quando se deseja definir o tipo de calendário para o seu arquivo. O GrADS sempre apresenta problemas de calendário, e esse operador é a solução para resolver isso.

As possibilidades são: **standard**, **proleptic_gregorian**, **360_day**, **365_day** e **366_day**.

Exemplo: **cdo setcalendar,standard if.nc of.nc**

6.6.3 Operador cname

Muda o nome da variável do arquivo.

Exemplo: **cdo cname,air,temp if.nc of.nc**

Muda o nome da variável **air** para **temp** e cria um novo arquivo **of.nc**.

6.6.4 Operador inverlat

Inverte a latitude. Se o seu dado está na forma N->S, este operador muda-o de S->N.

Exemplo: **cdo invertlat if.nc of.nc**

6.6.5 Operador para valores ausentes ou indefinidos

Fixa um novo valor ausente.

cdo setmissval,valor_indefinido if.nc of.nc

Em que **valor_indefinido** será o novo valor escolhido pelo usuário.

Exemplo: **cdo setmissval,-9999 if.nc of.nc**

O novo valor ausente/indefinido será -9999.

Fixa um intervalo de valores da variável para valores ausentes.

cdo setrtomiss,rmin,rmax if.nc of.nc

Não serão plotados os valores entre `rmin` e `rmax` porque eles são definidos como valores ausentes.

Exemplo: `cdo setrtomiss,0,5 if.nc of.nc`

O intervalo entre 0 e 5 não será plotado.

Fixa um intervalo de valores válido.

`cdo setvrange,rmin,rmax if.nc of.nc`

Esse operador apenas plota valores entre `rmin` e `rmax`.

Exemplo: `cdo setvrange,2,5 if.nc of.nc`

Apenas os valores entre 2 e 5 serão visualizados, os valores abaixo e acima desses limites são considerados indefinidos.

6.7 Operadores aritméticos

Avaliando expressões com o operador `expr`.

Esse operador é útil quando se deseja realizar algum cálculo, por exemplo, converter a temperatura de Kelvin para Celsius.

Exemplo: `cdo expr,'tc=air-273.15;' if.nc of.nc`

Em que `tc` (nome qualquer) é o nome da variável do arquivo que será criado (`of.nc`) e `air` é a variável do arquivo `if.nc`.

6.7.1 Operadores matemáticos

- `abs` (valor absoluto)
- `int` (apenas o valor inteiro)
- `nint` (inteiro mais próximo, faz arredondamento do valor)
- `pow` (potência, `cdo pow,2 if.nc of.nc`)
- `sqr` (eleva ao quadrado)
- `sqrt` (raiz quadrada)
- `exp` (exponencial)

- ln (log natural)
- log10 (log na base 10)
- sin (seno)
- cos (cosseno)
- tan (tangente)
- asin (arco seno)
- acos (arco cosseno)

Exemplo: **cdo nint if.nc of.nc**

6.7.2 Operações com constantes

- addc (soma a uma constante)
- subc (subtraí de uma constante)
- mulc (multiplica por uma constante)
- divc (divide por uma constante).

Exemplo: **Converte a temperatura de Kelvin para Celsius.**

cdo addc,-273.15 if.nc of.nc

ou

cdo subc,273.15 if.nc of.nc

Caso apareça a mensagem abaixo ao realizar a operação acima:

```
cdo subc (Warning): Some data values (min=-49720 max=-42867) are outside the
valid range (-32768 - 32767) of the used output precision!
```

Use the CDO option **-b 32** or **-b 64** to increase the output precision.

```
cdf_put_vara_double : ncid = 131072 varid = 5 val0 = -49338.000000
```

```
cdf_put_vara_double : varname = air
```

O CDO está dizendo para você utilizar a opção **-b 32** ou **-b 64**, e o comando ficará assim:

cdo -b 32 subc,273.15 if.nc of.nc

6.7.3 Operações usando dois conjunto de dados

Esses operadores realizam cálculos utilizando dados espaciais com as mesmas dimensões.

- add (soma dois campos)
- sub (subtrai dois campos)
- mul (multiplica dois campos)
- div (divide dois campos)
- min (mínimo de dois campos)
- max (máximo de dois campos)

Exemplo: `cdo add if1.nc if2.nc of.nc`

Soma os dois campos e guarda o resultado em of.nc

6.8 Cálculos estatísticos

Importante1: Caso seu arquivo possua dados indefinidos é melhor usar o `mean` (`fldmean`, `timmean`, dentre outros), caso contrário, use `avg` (`fldavg`, `timavg`, dentre outros). Se o dado não tem indefinido, a função `avg` é igual ao `mean`.

Importante2: Exemplo: Vamos usar os valores 2, 8, -999 e 4. Lembrando que -999 é um valor indefinido. Ao fazer a média usando o `mean`, o resultado será de 4,7, porque ele não considera o valor -999, o cálculo é feito apenas com $(2+8+4)/3 = 4,7$. Ao usar a `avg`, o resultado será -999, aqui será considerado o indefinido $(2 + 8 + (-999) + 4 = -999)$. O que isso significa? Caso a sua série de dados possua dados indefinidos, a sua série resultante terá valores indefinidos, por isso é importante ter informações sobre o dado para usar a função corretamente sem perda de dados.

Importante3: Como eu sei se o dado possui valores indefinidos? Basta usar o operador `infn` (`cdo infn if.nc`) e verificar a coluna onde tem Miss (quinta coluna do comando). O valor zero quer dizer que não há dados indefinidos.

6.8.1 Média de vários arquivos (ensemble)

Esse operador é útil quando se deseja realizar a média de vários arquivos ou ensemble.

Exemplo: `cdo ensmean if1.nc if2.nc if3.nc if4.nc if5.nc if6.nc of.nc`

ou

Exemplo: `cdo ensmean if[1-6].nc of.nc`

O CDO permite o uso de metacaracteres, por exemplo, `*`, `?` e `[]` dentre outros.

6.8.2 Campos bidimensionais

- `fldmin` (retorna o valor mínimo do domínio)
- `fldmax` (retorna o valor máximo do domínio)
- `fldsum` (retorna a soma do domínio)
- `fldmean` (retorna o valor médio do domínio)
- `fldavg` (retorna o valor médio do domínio)
- `fldvar` (retorna o valor da variância do domínio)
- `fldstd` (retorna o valor do desvio padrão do domínio).

Exemplo: `cdo fldmean if.nc of.nc`

Esse exemplo realiza a média espacial ou média na área do arquivo `if.nc`.

6.8.3 Cálculo estatístico zonal

- `zonmin` (para cada latitude o mínimo sobre todas as longitudes é calculado)
- `zonmax` (para cada latitude o máximo sobre todas as longitudes é calculado)
- `zonsum` (para cada latitude a soma sobre todas as longitudes é calculada)
- `zonmean` (para cada latitude a média sobre todas as longitudes é calculada)
- `zonavg` (para cada latitude a média sobre todas as longitudes é calculada)

- zonvar (para cada latitude a variância sobre todas as longitudes é calculada)
- zonstd (para cada latitude o desvio padrão sobre todas as longitudes é calculado)

Exemplo: **cdo zonmean if.nc of.nc**

6.8.4 Cálculo estatístico meridional

- mermin (para cada longitude o mínimo sobre todas as latitudes é calculado)
- mermax (para cada longitude o máximo sobre todas as latitudes é calculado)
- mersum (para cada longitude a soma sobre todas as latitudes é calculada)
- mermean (para cada longitude a média sobre todas as latitudes é calculada)
- meravg (para cada longitude a média sobre todas as latitudes é calculada)
- mervar (para cada longitude a variância sobre todas as latitudes é calculada)
- merstd (para cada longitude o desvio padrão sobre todas as latitudes é calculado)

Exemplo: **cdo mermean if.nc of.nc**

6.8.5 Cálculo estatístico vertical

- vertmin (extraí o valor mínimo de todos os níveis verticais)
- vertmax (extraí o valor máximo de todos os níveis verticais)
- vertsum (soma o valor de todos os níveis verticais)
- vertmean (média de todos os níveis verticais)
- vertavg (média de todos os níveis verticais)
- vertvar (variância de todos os níveis verticais)
- vertstd (desvio padrão de todos os níveis verticais)

Exemplo: **cdo vertmean if.nc of.nc**

6.8.6 Cálculo estatístico temporal

- `tinselmin` (mínimo temporal)
- `tinselmax` (máximo temporal)
- `tinselsum` (soma temporal)
- `tinselmean` (média temporal)
- `tinselavg` (média temporal)
- `tinselvar` (variância temporal)
- `tinselstd` (desvio padrão temporal)

Supondo que seu arquivo apresenta resolução temporal de meses (todos os meses completos, isto é, de janeiro até dezembro) com vários anos, para realizar uma média sazonal (MMA, JJA, SON e DJF), basta fazer de acordo com o comando abaixo. O número 3 significa fazer a média a cada três meses e o número 2 diz para pular apenas no início do sua série, ou seja, os meses de janeiro e fevereiro caso eles existam. Com esse comando, o resultado será uma média temporal sazonal.

Exemplo: `cdo tinselmean,3,2 if.nc of.nc`

6.8.7 Cálculo estatístico com média móvel

- `runmin` (média móvel mínima)
- `runmax` (média móvel máxima)
- `runsum` (média móvel soma)
- `runmean` (média móvel média)
- `runavg` (média móvel média)
- `runvar` (média móvel variância)
- `runstd` (média móvel desvio padrão)

Exemplo: `cdo runmean,4 if.nc of.nc`

Supondo a série temporal abaixo de comprimento 12:

13,3; 17,7; 9,9; 11,2; 1,7; 0,7; 1,1; 0,0; 2,0; 7,7; 9,2 e 14,1

O comando acima terá como resultado uma série de comprimento 9. O que foi feito?

Resolvendo o comando acima:

$$(13,3+17,7+9,9+11,2)/4=13,0$$

$$(17,7+9,9+11,2+1,7)/4=10,1$$

$$(9,9+11,2+1,7+0,7)/4=5,9$$

$$(11,2+1,7+0,7+1,1)/4=3,7$$

$$(1,7+0,7+1,1+0,0)/4=0,9$$

$$(0,7+1,1+0,0+2,0)/4=1,0$$

$$(1,1+0,0+2,0+7,7)/4=2,7$$

$$(0,0+2,0+7,7+9,2)/4=4,7$$

$$(2,0+7,7+9,2+14,1)/4=8,7$$

O resultado será: 13,0; 10,1; 5,9; 3,7; 0,9; 1,0; 2,7; 4,7 e 8,7.

6.8.8 Cálculo estatístico sobre todos os tempos

- `timmin` (valor mínimo)
- `timmax` (valor máximo)
- `timsun` (soma)
- `timmean` (média)
- `timavg` (média)
- `timvar` (variância)
- `timstd` (desvio padrão)

Exemplo: `cdo timmean if.nc of.nc`

Esse operador realiza a média temporal do arquivo `if.nc` e o resultado será apenas um tempo.

6.8.9 Cálculo estatístico diário

Converte dados para a resolução temporal diária.

- `daymin` (mínimo diário)
- `daymax` (máximo diário)
- `daysum` (soma diária)
- `daymean` (média diária)
- `dayavg` (média diária)
- `dayvar` (variância diária)
- `daystd` (desvio padrão diário)

Exemplo: `cdo daymean if.nc of.nc`

6.8.10 Cálculo estatístico mensal

Converte dados para a resolução temporal mensal.

- `monmin` (mínimo mensal)
- `monmax` (máximo mensal)
- `monsum` (soma mensal)
- `monmean` (média mensal)
- `monavg` (média mensal)
- `monvar` (variância mensal)
- `monstd` (desvio padrão mensal)

Exemplo: `cdo monmean if.nc of.nc`

6.8.11 Cálculo estatístico anual

Converte dados para a resolução temporal anual.

- `yearmin` (mínimo anual)
- `yearmax` (máximo anual)
- `yearsum` (soma anual)

- yearmean (média anual)
- yearavg (média anual)
- yearvar (variância anual)
- yearstd (desvio padrão anual)

Exemplo: **cdo yearmean if.nc of.nc**

6.8.12 Cálculo estatístico sazonal

Calcula os valores sazonais do arquivo de entrada.

- seasmin (mínimo sazonal)
- seasmx (máximo sazonal)
- seassum (soma sazonal)
- seasmean (média sazonal)
- seasavg (média sazonal)
- seasvar (variância sazonal)
- seasstd (desvio padrão sazonal)

Exemplo: **cdo seasmean gpcp.as.2000.2001.nc sazonal.nc**

Ao utilizar este operador considerando que o seu arquivo gpcp.as.2000.2001.nc possua todos os meses (jan, ..., dez) aparecerão as seguintes mensagens de aviso, não precisa se preocupar pois o cálculo foi feito corretamente:

```
cdo seasmean (Warning): Season 1 ( 2000-01-01) has only 2 input time steps!
cdo seasmean (Warning): Season 9 ( 2001-12-01) has only 1 input time step!
```

Faz todo o sentido porque ele tenta fazer a média do mês de dezembro de 1999 (que não existe), janeiro e fevereiro de 2000, por isso ele diz que a primeira estação tem apenas dois tempos (média de janeiro e fevereiro de 2000). Para a última estação, o raciocínio é o mesmo, ou seja, somente há o mês de dezembro, logo não é possível realizar a média, por isso, esse valor é repetido.

6.8.13 Valor estatístico mensal de vários anos

Este módulo cria climatologia. O arquivo de saída terá 12 tempos (jan, fev, ..., dez). Válido para dados horários, diários e mensais. Aqui está sendo mostrado apenas o mensal.

- ymonmin (mínimo mensal)
- ymonmax (máximo mensal)
- ymonsum (soma mensal)
- ymonmean (média mensal)
- ymonavg (média mensal)
- ymonvar (variância mensal)
- ymonstd (desvio padrão mensal)

Exemplo: `cdo ymonmean if.nc of.nc`

O comando acima criará a climatologia.

6.8.14 Valor estatístico sazonal de vários anos

Este módulo cria climatologia sazonal. O arquivo de saída terá 4 tempos (DJF, MAM, JJA e SON). O primeiro tempo corresponde ao verão (DJF), o segundo ao outono (MAM), o terceiro ao inverno (JJA) e o quarto a primavera (SON). O cálculo utiliza todos os meses disponíveis para realizar a média sazonal. Por exemplo, para DJF serão considerados todos os meses janeiro, fevereiro e dezembro. O mesmo se aplica para as demais estações.

- yseasmin (mínimo sazonal)
- yseasmax (máximo sazonal)
- yseassum (soma sazonal)
- yseasmean (média sazonal)
- yseasavg (média sazonal)

- yseasvar (variância sazonal)
- yseasstd (desvio padrão sazonal)

Exemplo: `cdo yseasmean if.nc of.nc`

6.9 Interpolação

6.9.1 Operador remapbil

Realiza interpolação bilinear, porém há outras possibilidades de interpolação.

Exemplo: Interpolando o dado do modelo (pr.CCSM4.1996.2005.nc) para a resolução do dado do GPCP (pr.GPCP.1996.2005.nc).

`cdo remapbil,pr.GPCP.1996.2005.nc pr.CCSM4.1996.2005.nc ccsm4.int.nc`

Compare o resultado com o operador `griddes`.

```
cdo griddes ccsm4.int.nc
cdo griddes pr.GPCP.1996.2005.nc
```

6.10 Importação e exportação

6.10.1 Importação de conjunto de dados binários

Converte um arquivo binário para NetCDF dado o arquivo descritor (.ctl) do binário.

Exemplo: `cdo -f nc import__binary if.ctl of.nc`

`-f nc` = converte para NetCDF

`if.ctl` = arquivo descritor do arquivo binário

Importante: A resolução temporal tem que ser linear, isto é, sem data faltante. O CDO converte o dado para NetCDF até onde o dado está completo.

6.10.2 Conversão de arquivo texto para NetCDF

Converte um arquivo ASCII em NetCDF.

`cdo -f nc input,r1x1 of.nc < arquivo.txt`

`r1x1` = o arquivo `arquivo.txt` contém 12 linhas, isto é, 12 meses para um dado

pontual (NX x NY). Esse arquivo poderia ser horário, diário, mensal ou anual.

of.nc = nome do arquivo de saída

Ao gerar o arquivo **of.nc** abra-o no GrADS.

```
ga-> sdfopen of.nc Scanning self-describing file: of.nc SDF file has no discernable
time coordinate - using default values. gadsdf: SDF file does not have any non-
coordinate variables.
```

O GrADS dirá que o dado não tem coordenada de tempo associada ao arquivo, e para resolver isso, basta fazer:

```
cdo -r settaxis,2000-01-01,12:00:00,1mon if.nc of1.nc
```

-r = adiciona um eixo de tempo relativo.

2000-01-01,12:00:00 = data que o usuário seleciona com base no seu arquivo

1mon = o dado é mensal, isso corresponde ao intervalo de tempo (dt)

6.10.3 Extrair arquivos ASCII de NetCDF

a) Exemplo a): Saída bruta, sem formatação

```
cdo output if.nc > output.txt
```

b) Exemplo b): Saída processada, com formatação

```
cdo outputf,%6.2f if.nc > output.txt
```

%6.2f = 6 elementos, incluindo ponto e sinal (+ ou -) com duas casas decimais

6.10.4 Correlação

O CDO possui dois operadores para calcular a correlação, são eles: fldcor (correlação espacial) e o timcor (correlação temporal). Os arquivos têm que possuir as mesmas dimensões para utilizar esses operadores.

6.10.5 Correlação espacial

O operador fldcor correlaciona todos os pontos de grade para cada tempo das duas variáveis. O resultado é uma série temporal da correlação entre elas.

Exemplo prático deste operador:

Será feita a correlação entre a precipitação e a temperatura do ar em 1000hPa. Os dados mensais correspondem ao período de 2005 a 2007 sobre América do Sul.

Dado de precipitação: `prec.2005.2007.nc`

Dado de temperatura do ar em 1000hPa: `tar.1000hpa.2005.2007.nc`

E finalmente, basta digitar o comando abaixo:

```
cdo fldcor prec.2005.2007.nc tar.1000hpa.2005.2007.nc corr.prec.temp.nc
```

A variável do arquivo `corr.prec.temp.nc` se chama `precip` porque o CDO guarda a informação do nome da variável do arquivo `prec.2005.2007.nc`.

Para mudar esse nome, basta utilizar o operador `chname`.

```
cdo chname,precip,corr corr.prec.temp.nc correlacao.nc
```

Agora, o arquivo `correlacao.nc` possui o nome `corr` segundo o operador `chname`.

6.10.6 Correlação temporal

Diferente do operador `fldcor`, o `timcor` realiza a correlação em cada ponto de grade para todos os tempos das duas variáveis. O resultado é um arquivo espacial da correlação com apenas um tempo.

Utilizando os arquivos de precipitação e temperatura do ar, será calculada a correlação no tempo da seguinte forma:

```
cdo timcor prec.2005.2007.nc tar.1000hpa.2005.2007.nc corr.nc
```

A variável do arquivo `corr.prec.temp.nc` se chama `precip`, para alterar o nome da variável caso seja necessário, utilize o operador `chname`.

```
cdo chname,precip,corr corr.nc correlacao.tempo.nc
```

7 MÓDULO PRÁTICO

A ideia deste tópico é mostrar os mais variados exemplos de uso do CDO. Eles são adaptações da internet, dúvidas de colegas e de minha própria autoria.

7.1 Alterando a coordenada vertical

A dica abaixo serve para qualquer situação em que a coordenada vertical do dado esteja em Pascal (Pa).

Os níveis verticais dos modelos do CMIP5 estão em Pa, e normalmente os cálculos são feitos utilizando hectopascal (hPa). Será utilizado o operador **setzaxis** para realizar essa alteração de Pa para hPa.

O arquivo que será utilizado se chama **cmip5.nc**.

Digite o comando abaixo no seu terminal Linux para ver a descrição da coordenada vertical do arquivo cmip5.nc.

cdo zaxisdes cmip5.nc

O resultado será:

```
#  
# zaxisID 1  
#  
zaxistype = pressure  
size = 8 ⇒ o arquivo possui 8 níveis verticais  
name = plev  
longname = pressure  
units = Pa ⇒ a unidade do nível vertical está em Pa  
levels = 100000 85000 70000 50000 25000 10000 5000 1000 ⇒ os níveis verticais em  
Pa, são 8 no total  
bounds = 107500-92500 92500-77500 77500-60000 60000-37500 37500-17500 17500-  
7500 7500-3000 3000-1000
```

Segue a dica:

Crie um arquivo texto chamado nivel, e adicione as 6 linhas abaixo. Esse arquivo será lido pelo operador setzaxis. A unidade e a coordenada vertical está em hPa.

```
zaxistype = pressure
size = 8
name = lev
longname = pressure
units = hPa
levels = 1000 850 700 500 250 100 50 10
```

Utilize o comando abaixo:

```
cdo setzaxis,nivel cmip5.nc cmip5_novo.nc
```

Esse comando **altera a estrutura** do arquivo cmip5.nc e salva as alterações em cmip5_novo.nc.

Ao digitar o comando

```
cdo zaxisdes cmip5_novo.nc
```

As novas alterações são visualizadas logo abaixo. Agora, o arquivo cmip5_novo.nc apresenta a unidade e a coordenada vertical em hPa.

```
#
# zaxisID 1
#
zaxistype = pressure
size = 8
name = lev
longname = pressure
units = hPa
levels = 1000 850 700 500 250 100 50 10
```

7.2 Alterando os valores NaN do arquivo NetCDF

Alguns arquivos NetCDF possuem o valor NaN, como por exemplo, no arquivo prec.xavier.nc.

Ao digitar o comando:

```
cdo infon prec.xavier.nc
```

Serão retornadas as seguintes informações:

-1 :	Date	Time	Level	Gridsize	Miss :	Minimum	Mean	Maximum :	Parameter name
1 :	2008-01-01	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
2 :	2008-01-02	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
3 :	2008-01-03	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
4 :	2008-01-04	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
5 :	2008-01-05	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
6 :	2008-01-06	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
7 :	2008-01-07	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
8 :	2008-01-08	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
9 :	2008-01-09	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec
10 :	2008-01-10	00:00:00	0	27216	0 :	-nan	-nan	-nan :	prec

Isso gera problemas ao visualizar a variável e isso ocorre porque ela não tem o atributo `_FillValue` e `missing_value`. Os atributos dessa variável foram vistos com o comando abaixo:

```
ncdump -h prec.xavier.nc
```

Será mostrado apenas um pequeno trecho do comando acima:

```
float prec(time, latitude, longitude) ;
    prec:standard_name = "prec" ;
    prec:units = "mm" ;
```

Mas isso pode ser contornado com o comando abaixo:

```
cdo -L -setmissval,-999.9 -setmissval,nan prec.xavier.nc prec.nc
```

O que foi feito? Inicialmente, os valores nan foram transformados em valores ausentes com o operador `setmissval` e posteriormente, esse valor ausente foi substituído pelo novo valor ausente de `-999.9`.

7.3 Alterando valores do arquivo NetCDF

Supondo que o usuário tenha um mapa de vegetação com diversas classes, e o mesmo deseja alterar algumas dessas classes. Isso pode ser feito com o comando abaixo:

```
cdo setvals,1,6 vegetacao.nc vegetacao.alterado.nc
```

Em que **1** é a classe antiga e **6** é a classe nova. Todos os **pontos de grade do arquivo** com a classe 1 serão substituídos pelo valor 6.

vegetacao.nc é o arquivo de entrada e **vegetacao.alterado.nc** é o arquivo de saída.

Caso seja necessário alterar mais de uma classe, continue com o mesmo raciocínio

da linha acima:

```
cdo setvals,1,6,3,7,8,12 vegetacao.nc vegetacao.alterado.nc
```

Nessa caso em particular, **1** será trocado pelo valor **6**, **3** será trocado pelo **7** e **8** pelo **12**.

Para alterar um intervalo de valores para um único valor:

Por exemplo, fixar o intervalo de valores entre 16 e 18 para 20. Proceda da seguinte forma:

```
cdo setrtoc,16,18,20 if.nc of.nc
```

Entendendo o comando, **16** e **18** é o intervalo que será substituído pelo valor **20**. Isso dependerá da variável utilizada. Altere de acordo com suas necessidades.

if.nc é o arquivo de entrada e **of.nc** é o arquivo de saída.

7.4 Anomalia climatológica zonal de altura geopotencial

Para quem quiser saber o que isso representa, veja o artigo (Figuras 8 e 9) de [Cavalcanti et al. \(2002\)](#).

Removendo a média zonal do dado:

```
cdo zonmean hgt.200hpa.1991.2005.nc tmp.01.nc
```

Foi gerado o arquivo **tmp.01.nc**.

Deixando o dado de média zonal compatível (operador `enlarge`) com o dado espacial de altura geopotencial:

```
cdo enlarge,hgt.200hpa.1991.2005.nc tmp.01.nc tmp.02.nc
```

O arquivo **tmp.01.nc** será “expandido” para ter as mesmas dimensões de **hgt.200hpa.1991.2005.nc**, e finalmente será gerado o arquivo **tmp.02.nc**.

Calculando a anomalia zonal de altura geopotencial. A saída será mensal.

```
cdo sub hgt.200hpa.1991.2005.nc tmp.02.nc tmp.03.nc
```

Calculando a média sazonal caso queira a saída por estações do ano:

```
cdo yseasmean tmp.03.nc anomalia.zonal.sazonal.ncep.nc
```

7.5 Calculando a relação entre os desvios padrão simulado e observado

A razão é calculada entre o desvio padrão observado (σ_{obs}) e simulado (σ_{mod}).

$$\sigma_{obs} = \sqrt{(n-1)^{-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

$$\sigma_{mod} = \sqrt{(n-1)^{-1} \sum_{i=1}^n (y_i - \bar{y})^2}$$

$$R_{\sigma} = \frac{\sigma_{obs}}{\sigma_{mod}}$$

Os dois arquivos utilizados são mensais e possuem 12 meses cada um.

Os dados utilizados neste exemplo serão: pr.media.GPCP.nc e pr.media.CCSM4.nc

A razão é calculada da seguinte forma:

```
cdo -s -output -div -timstd1 pr.media.GPCP.nc -timstd1  
pr.media.CCSM4.nc > rdesv.pad.txt
```

O resultado é adimensional. Quanto mais próximo do valor 1, maior será a semelhança entre as séries.

7.6 Calculando a diferença entre o tempo posterior e o anterior

Em algumas situações é necessário calcular a diferença entre os tempos t2 (tempo posterior) e t1 (tempo anterior). Por exemplo, o arquivo prec.2013.CRU.mensal.nc possui 12 tempos (mensal) para a variável precipitação. O que desejamos calcular é a diferença entre os tempos t2 e t1 para todos os 12 tempos do arquivo. O resultado será um arquivo com o número de tempos nt-1, ou seja, 12-1=11 tempos, para este exemplo. O nt representa o número de tempos do seu arquivo.

O dado utilizado será o prec.2013.CRU.mensal.nc.

Para saber o número total de tempos do arquivo:

```
cdo ntime prec.2013.CRU.mensal.nc
```

O valor retornado será 12 tempos ou meses porque o dado é mensal.

Para realizar o cálculo da diferença entre os tempos, basta utilizar o comando abaixo. No denominador basta inserir o valor obtido com o ntime (12):

```
cdo -sub -seltimestep,2/12 prec.2013.CRU.mensal.nc  
prec.2013.CRU.mensal.nc dif.nc
```

O 2/12 quer dizer, do tempo 2 ao 12, ou 2, 3, 4, ...,12.

7.7 Calculando a anomalia padronizada

Para calcular a anomalia padronizada, utiliza-se o conjunto de comandos abaixo:

O dado utilizado será o `prec.nc`.

```
cdo sub prec.nc -timmean prec.nc anom.nc  
cdo div anom.nc -timstd anom.nc anom.padronizada.nc
```

7.8 Calculando a velocidade do vento

A velocidade a partir das componentes do vento é calculada da seguinte maneira:

$$velocidade = \sqrt{u^2 + v^2}$$

7.8.1 Método 1

Outra forma de realizar essa tarefa é por meio do encadeamento de operadores:

```
cdo -s -O -sqrt -add -sqr uwnd.1000hpa.2010.nc -sqr  
vwnd.1000hpa.2010.nc velocidade.nc
```

Lembrando que a execução dos comandos será feita da direita para esquerda.

Caso seja necessário alterar o nome da variável do arquivo `velocidade.nc`, proceda da seguinte forma:

```
cdo chname,uwnd,vel velocidade.nc nova.velocidade.nc
```

`chname` altera o nome `uwnd` para `vel`.

A opção `-s` não mostra as mensagens de processamento e `-O` sobrescreve o arquivo caso ele exista.

7.8.2 Método 2

Outra forma mais fácil de realizar esse cálculo é utilizando o operador `expr`.

```
cdo -setunit,'m/s' -expr,'vel=sqrt(u*u+v*v)' if.nc of.nc
```

Em que:

setunit cria a unidade (m/s)

vel é o nome da nova variável que será criada dentro do arquivo de saída **of.nc**.

u e **v** são as componentes zonal e meridional do vento do seu arquivo de entrada **if.nc**.

Importante: As variáveis **u** e **v** precisam estar no mesmo arquivo, não podem estar em arquivos separados. Caso isso ocorra, utilize o operador **merge** para juntá-los.

7.9 Calculando o erro absoluto médio (EAM)

O EAM é calculado de acordo com a equação abaixo.

$$EAM = \frac{\sum_{i=1}^n |S_i - O_i|}{n}$$

Em que S_i e O_i são os valores simulados e observados, respectivamente e n é o comprimento da amostra.

Os dois arquivos são mensais e possuem 12 meses cada um.

Os dados utilizados neste exemplo serão: **pr.media.GPCP.nc** e **pr.media.CCSM4.nc**

Para saber o número total de tempos do arquivo, utiliza-se o comando **ntime**.

Exemplo: `cdo ntime pr.media.GPCP.nc`

O valor retornado será 12.

E para calcular o EAM utiliza-se o comando abaixo:

```
cdo -s -output -divc,12 -abs -timsum -sub pr.media.CCSM4.nc  
pr.media.GPCP.nc > EAM.txt
```

O resultado tem a mesma unidade do dado utilizado, neste caso, o resultado será em mm/dia. Quanto menor o valor, maior será a semelhança entre as séries.

7.10 Calculando o número de dias consecutivos de precipitação

Como calcular o número de dias consecutivos de precipitação acima de um dado limiar? Por exemplo, o número de dias consecutivos em que a precipitação estava

acima de 4 mm/dia.

O dado utilizado neste exemplo será: `prec.diario.2010.nc`.

```
cdo -consects -gtc,4 prec.diario.2010.nc prp.nc
```

Para ver os valores do arquivo de saída, digite o comando:

```
cdo infon prp.nc
```

7.11 Calculando o bias

O bias é calculado de acordo com a equação abaixo.

$$bias = \frac{\sum_{i=1}^n (S_i - O_i)}{n}$$

Em que S_i e O_i são os valores simulados e observados, respectivamente e n é o comprimento da amostra.

Os dois arquivos são mensais e possuem 12 meses cada um.

Os dados utilizados neste exemplo serão: `pr.media.GPCP.nc` e `pr.media.CCSM4.nc`

Para saber o número total de tempos do arquivo, utiliza-se o comando `ntime`.

```
Exemplo: cdo ntime pr.media.GPCP.nc
```

O valor retornado será 12.

E para calcular o bias com o CDO, utiliza-se o comando abaixo:

```
cdo -s -output -divc,12 -timsum -sub pr.media.CCSM4.nc  
pr.media.GPCP.nc > bias.txt
```

Em que 12 representa o número de meses do dado.

O resultado tem a mesma unidade do dado utilizado, neste caso, o resultado será em mm/dia. Valores positivos (negativos) indicam superestimativa (subestimativa) e quanto mais próximo de zero, menor será o erro.

7.12 Calculando média espacial sobre a Bacia Amazônica

Será explicado como obter a média espacial de precipitação somente sobre a Bacia Amazônica. O arquivo dessa bacia possui resolução espacial de 25km enquanto que

o dado de precipitação apresenta 2.5°.

Os arquivos utilizados correspondem ao contorno da Bacia Amazônica (bacia.amazonica.25km.nc) e ao dado de precipitação (gpcp.2012.nc).

Inicialmente, o arquivo bacia.amazonica.25km.nc será interpolado para a mesma resolução do dado de precipitação (gpcp.2012.nc) com o operador remapbil e será gerado o arquivo mascara.nc que contém o contorno da bacia na mesma resolução do dado de precipitação.

```
cdo remapbil,gpcp.2012.nc bacia.amazonica.25km.nc mascara.nc
```

Feito isso, agora será necessário mascarar o dado de precipitação apenas no domínio da bacia, com o comando abaixo:

```
cdo ifthen mascara.nc gpcp.2012.nc tmp.nc
```

O arquivo tmp.nc já está mascarado e há somente informações de precipitação dentro do contorno da bacia.

Para realizar a média espacial, será utilizado o operador fldmean e o outputf para formatar a saída.

```
cdo -outputf,%4.1f -fldmean tmp.nc > prec.bac.amaz.txt
```

O “4.1f” é a formatação da saída com 4 valores e uma casa decimal. O arquivo prec.bac.amaz.txt possui os valores de precipitação no domínio da Bacia Amazônica.

7.13 Calculando RMSE

O RMSE é calculado da seguinte forma:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (S_i - O_i)^2}{n}}$$

Em que S_i e O_i são os valores simulados e observados, respectivamente e n é o comprimento da amostra.

Os dados utilizados neste exemplo serão: pr.GPCP.2005.nc e pr.CCSM4.2005.nc.

E para calcular o RMSE, basta digitar os comandos abaixo:

```
cdo -s output -sqrt -divc,10368 -fldsum -sqr -sub pr.CCSM4.2005.nc
```

pr.GPCP.2005.nc > rmse.txt

Em que 10368 refere-se ao número de pontos de grade do domínio utilizado. Caso seu dado possua dados ausentes, basta subtrair esse valor pelo total de dados ausentes do seu arquivo.

O resultado tem a mesma unidade do dado utilizado, neste caso, o resultado será em mm/dia. Quanto menor o valor, maior será a semelhança entre as séries.

Como saber o total de pontos de grade do meu arquivo? Basta digitar o comando:

```
cdo infon pr.CCSM4.2005.nc
```

ou

```
cdo infon pr.GPCP.2005.nc
```

E o resultado mostrará uma coluna com o nome de **Gridsize** que equivale a 10368. E a coluna de dados ausentes é representada pelo nome **Miss** e neste caso, vale zero. Logo, não há necessidade de subtrair o valor 10368 de zero.

7.14 Calculando “pentadas”

Agradecimentos ao Rômulo Oliveira (rom.aug9@gmail.com) pela versão original do script. A ideia original é dele, apenas fiz algumas adaptações.

Para calcular pentadas, utilize o script abaixo. Para torná-lo executável, utilize o comando **chmod +x script.sh**. E para executar, basta digitar **./script.sh**.

Esse script a partir de uma determinada data calcula a pentada, somente isso.

O arquivo que será utilizado se chama: `prec.sa.2010.nc`.

```
#!/bin/bash
```

```
dt=4 # comprimento da “pentada”.
```

```
fin="prec.sa.2010.nc" # nome do arquivo de entrada
```

```
lon="-55" # longitude
```

```
lat="-5" # latitude
```

```

rm -f pentada.txt # remove o arquivo de saída caso ele exista

for i in "2010-10-01" "2010-12-10" ; do # datas para calcular a pentada

diai=$(date --date="$i -$dt days" +%Y-%m-%d) # dia inicial

diaf=$i # dia final

fout="lixo.$diai.$diaf.txt" # nome do arquivo de saída

echo $diai $diaf

cdo -s -O -outputcenter -remapnn,lon="$lon" _lat="$lat" -timsun -seldate,$diai"T00:00:00,"$diaf"T00:00:00" $fin
> $fout

sed -i '13!d' $fout

echo $diai $diaf > tmp1.txt

paste tmp1.txt $fout >> tmp2.txt

cat tmp2.txt | sed 's/[t]/ /g' | tr -s ' ' | sed 's/ /;/g' > pentada.txt

rm -f $fout tmp1.txt tmp2.txt

done

# Fim do script

```

O arquivo de saída **pentada.txt** tem o formato abaixo que corresponde a data inicial e final, longitude, latitude e precipitação. O separador utilizado é o ponto e vírgula.

```

2010-09-27;2010-10-01;-55;-5;9.06667
2010-12-06;2010-12-10;-55;-5;45.52

```

Outra forma de calcular pentadas pode ser feita com o operador `timselmean`. O comando abaixo gera um arquivo com 73 tempos (01 janeiro a 31 de dezembro = 365 dias/5 = 73 pentadas ou tempos).

Para esse exemplo será utilizado o arquivo `prec.2009.nc`.

Exemplo1: `cdo timselmean,5 prec.2009.nc pentada.nc`

A diferença entre o script que foi feito e o operador `tinselmean` é que no script é passado um dia qualquer e assim é feita a média. Por outro lado, o operador `tinselmean` calcula a pentada a cada 5 dias (neste exemplo) independente do dia selecionado.

Exemplo2: Média corrida de comprimento 5 (dias).

```
cdo runmean,5 prec.2009.nc med.corrida.nc
```

7.15 Convertendo um arquivo de perfil vertical formato texto para NetCDF

A partir da versão 1.7.2 é possível gerar perfil vertical por meio de um arquivo texto. Por exemplo, uma sondagem para o formato NetCDF. Para realizar essa tarefa basta utilizar o comando abaixo:

```
cdo -f nc input,r1x1,zaxis temp_z.nc < temp_z.txt
```

Em que `zaxis` é um arquivo texto que o usuário deverá criar e contém a descrição da coordenada vertical do dado no seguinte formato:

```
zaxistype = pressure
size = 462
units = hPa
levels =
1008.1
1001.1
995.20
989.30
983.40
```

Em que:

`zaxistype = pressure` ⇒ **Nome da coordenada vertical.**

`size = 462` ⇒ **Quantos níveis verticais serão criados.**

`units = hPa` ⇒ **Unidade do nível vertical.**

`levels =` ⇒ **Os níveis de interesse. São mostrados apenas alguns níveis, mas deveriam ter 462 níveis.**

O `temp_z.nc` é o arquivo que será criado e `temp_z.txt` é o arquivo que contém os valores de temperatura na vertical. Esse comando deve ser utilizado para cada

coluna do arquivo, não tente ler a sondagem de uma vez, por isso, separe o arquivo em colunas.

7.16 Criando anomalias

Para o cálculo de anomalia, utilize o comando abaixo:

```
cdo -ymonsub if.nc -ymonmean if.nc of.nc
```

Exemplo: Anomalia mensal de Radiação de Onda Longa (ROL). Proceda da seguinte forma:

```
cdo -b 32 -ymonsub rol.2000.2004.nc -ymonmean rol.2000.2004.nc  
anom.rol.nc
```

Com o comando acima, será criado o arquivo de anomalia mensal **anom.rol.nc**.

7.17 Criando climatologia

Será utilizado o operador **ymonmean**.

```
cdo ymonmean gpcp.2000.2009.nc clima.prec.nc
```

Com isso, será gerado o arquivo **clima.prec.nc** com 12 tempos (use o cdo **ntime clima.prec.nc** para ver o total de tempos desse arquivo).

7.18 Criando máscara para oceano ou continente

O dado utilizado neste exemplo será: **gpcp.2012.nc**

Primeiramente, criando a máscara na mesma resolução do dado **gpcp.2012.nc**

```
cdo -f nc -remapnn,gpcp.2012.nc -gtc,0 -topo mascara.nc
```

Mascarando o oceano

```
cdo ifthen mascara.nc gpcp.2012.nc ocean_mask_prec.nc
```

Mascarando o continente

```
cdo ifnotthen mascara.nc gpcp.2012.nc land_mask_prec.nc
```

7.19 Extrairindo a série temporal de um ponto com diversas informações

```
cdo -outputtab,date,lat,lon,value -remapnn,lon=270_lat=-2 seu.arquivo.nc >
serie.txt
```

Em que:

date,lat,lon,value referem-se as colunas de data, valor da latitude, valor da longitude, valor da variável nesse ponto de latitude e longitude.

O valor **270** (longitude) e **-2** (latitude) são as coordenadas desejadas para extrair a série.

O **seu.arquivo.nc** corresponde ao arquivo de entrada e **serie.txt** é o arquivo texto que armazenará os valores.

O resultado de um exemplo está logo abaixo:

```
cdo -outputtab,date,lat,lon,value -remapnn,lon=270_lat=-2 ps.nc >
psfc.txt
```

```
1979-01-15  -2  270  1010.48
1979-02-13  -2  270  1010.12
1979-03-16  -2  270  1009.87
```

Em que a coluna 1 = data, coluna 2 = latitude, coluna 3 = longitude e coluna 4 = valor no ponto -2 (lat) e 270 (lon).

7.20 Extrairindo apenas a série temporal de um ponto

```
cdo -output -remapnn,lon=-60_lat=-10 if.nc > serie.txt
```

lon=-60 e **lat=-10** representam a longitude e latitude de interesse, respectivamente.

if.nc é o arquivo de entrada.

serie.txt é o arquivo com a série temporal.

7.21 Interpolando mapa de vegetação

Link para o dado original na resolução de 10km:

http://webmap.ornl.gov/ogcdown/wcsdown.jsp?dg_id=10004_31

Relembrando que há 360° de longitude, então basta dividir pela resolução desejada, isto é, 0.5° para obter 720 pontos. Para latitude, o raciocínio é o mesmo, ou seja, 180° de latitude, logo $180^\circ/0.5^\circ$ é igual a 360 pontos de latitude.

O comando abaixo interpolará o mapa de vegetação do IGBP com 17 classes na resolução espacial de 10km para 50km ou 0.5° .

```
cdo remaplaf,r720x360 IGBP.10km.nc IGBP.50km.nc
```

Ao digitar o comando:

```
cdo griddes IGBP.50km.nc
```

Serão retornadas as informações abaixo. O que está em negrito (**xinc** e **yinc**) representa a nova resolução espacial de 50km.

```
#  
# gridID 1  
#  
gridtype = lonlat  
gridsize = 259200  
xname = lon  
xlongname = longitude  
xunits = degrees_east  
yname = lat  
ylongname = latitude  
yunits = degrees_north  
xsize = 720  
ysize = 360  
xfirst = 0  
xinc = 0.5  
yfirst = -89.75  
yinc = 0.5
```

7.22 Mascando regiões

Pode-se mascarar regiões usando operador **maskregion**, como no exemplo abaixo:

Primeiro, deve-se criar um arquivo texto com o domínio a ser mascarado. Por exem-

plo, vamos mascarar o domínio entre as longitude -80 e -40 e latitude entre -10 e +10. O arquivo texto deverá conter as seguintes informações:

```
-80 10
-80 -10
-40 -10
-40 10
```

Salve o arquivo com o nome de **mask** com as informações acima.

Em seguida, use o comando abaixo:

```
cdo maskregion,mask seu.arquivo.nc saida.nc
```

Em que:

seu.arquivo.nc é o arquivo da entrada e **saida.nc** é o arquivo de saída com a região mascarada.

Ou simplesmente utilize o comando:

```
cdo masklonlatbox,-80,-40,10,-10 seu.arquivo.nc saida.nc
```

7.23 Mascarando valores de velocidade e vetor do vento

A sugestão foi proposta pelo Cristiano Prestelo (<http://prestelocristiano.blogspot.com.br>).

Serão utilizados os arquivos **uwnd.nc** e **vwnd.nc**.

Supondo que as variáveis da componente zonal (**u**) e meridional (**v**) estão em arquivos separados. Inicialmente, vamos juntá-las em um único arquivo para calcular a velocidade do vento.

```
cdo merge uwnd.nc vwnd.nc uv.nc
```

O próximo passo é calcular a velocidade do vento. Será gerado um novo arquivo em que a variável se chamará **vel**.

```
cdo expr,'vel=sqrt(uwnd*uwnd+vwnd*vwnd);' uv.nc vel.nc
```

Será criada a máscara de velocidade, para isso, basta definir o intervalo de velocidade do seu interesse. Será utilizado o intervalo entre 0 e 4 m s⁻¹. Em seguida, o operador

setrtomiss defini valores ausentes ou indefinidos para um dado intervalo de valores, neste caso, entre 0 e 4.

cdo setrtomiss,0,4 vel.nc mascara.nc

O arquivo `mascara.nc` será utilizado para mascarar as componentes zonal (u) e meridional (v) do vento com o uso do operador **ifthen**.

cdo ifthen mascara.nc uwnd.nc vento.u.nc ⇒ mascarando valores da componente zonal

cdo ifthen mascara.nc vwnd.nc vento.v.nc ⇒ mascarando valores da componente meridional

E finalmente, basta visualizar a velocidade e o vetor do vento (arquivos `vento.u.nc` e `vento.v.nc`) que eles apresentarão somente valores acima de 4 m s^{-1} .

7.24 Preenchimento de dados ausentes

Essa dica foi proposta pelo Augusto Veiga (<https://scientificmet.wordpress.com>).

O arquivo que será utilizado se chama `zg_hadgem2_200501.nc`.

O operador `fillmiss` é capaz de realizar o preenchimento de dados faltantes mediante interpolação bilinear dos vizinhos mais próximos.

cdo fillmiss zg_hadgem2_200501.nc geo.nc

Os dados que antes eram ausentes, agora foram preenchidos com esse operador.

7.25 Redução de dimensão de arquivos netCDF

É possível reduzir o número de dimensões com a opção `--reduce_dim`. Ela somente funciona com os operadores `fldmin`, `fldmax`, `fldsum`, `fldmean`, `fldavg`, `fldstd`, `fldstd1`, `fldvar`, `fldvar1`, `fldpctl`, `fldcor`, `fldcovar`, `timpctl`, `timselpctl`, `timsort`, `timselmin`, `timselmax`, `timselsum`, `timselmean`, `timselavg`, `timselvar`, `timselvar1`, `timselstd`, `timselstd1`, `timmin`, `timmax`, `timsun`, `timmean`, `timavg`, `timvar`, `timvar1`, `timstd`, `timstd1`, `timcor` e `timcovar`.

Exemplo: O arquivo `prec.2009.nc` possui três dimensões, isto é, `time`, `lat` e `lon` conforme a informação abaixo:

float PREC(time, lat, lon) ⇒ visto com o comando `ncdump -h prec.2009.nc`.

E representa um dado espacial. Para realizar a redução das dimensões latitude e longitude basta usar o comando abaixo:

```
cdo --reduce_dim -fldmean prec.2009.nc reducao.nc
```

O `fldmean` realiza a média espacial o que resulta em apenas um ponto de lat e um ponto de lon. O `--reduce_dim` apaga as dimensões lat e lon sendo o resultado armazenado em `reducao.nc`.

Ao digitar o comando:

```
ncdump -h reducao.nc
```

O resultado será:

```
float PREC(time)
```

Isto é, houve redução das duas dimensões (lat e lon) para apenas uma (time).

8 Links interessantes

- Página que contém esse tutorial: <https://sites.google.com/site/jgmsantos/tutoriais/cdo>
- Página oficial do CDO: <https://code.zmaw.de/projects/cdo>
- Manual do CDO: <https://code.zmaw.de/projects/cdo/wiki/Cdo#Documentation>
- Lista de discussão: <https://code.zmaw.de/projects/cdo/boards>

REFERÊNCIAS BIBLIOGRÁFICAS

MAX-PLANCK-INSTITUT FÜR METEOROLOGIE. **Climate Data Operators**.
Disponível em: <<https://code.zmaw.de/projects/cdo/wiki/Cdo#Documentation>>.
Acesso em: 02 de abr. 2014.