

MeteoLab

一个用于气象数据诊断分析的 Matlab 工具箱

Dy

2022 年 4 月

目 录

一、MeteoLab 简介	3
二、安装步骤	3
三、函数使用说明	3
2.1 基本常数	3
2.2 基础物理量	3
2.3 差分/梯度/平滑	4
2.4 动力相关量	5
2.5 热力相关量	5
2.6 水汽相关量	6
四、高级功能	6
1. Shuman-Shapiro 滤波	6
2. 涡度收支诊断	6
3. 台风诊断	6

MeteoLab 使用说明

一、MeteoLab 简介

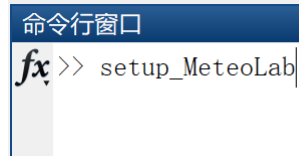
MeteoLab 是基于 Matlab 开发的，专门用于气象数据诊断和分析的工具箱，包含气象物理量诊断相关的基本函数，可快速计算出所需的气象物理量。可用于气象科研、数据可视化等领域。需要在 Matlab2016a 以上版本运行。

二、安装步骤

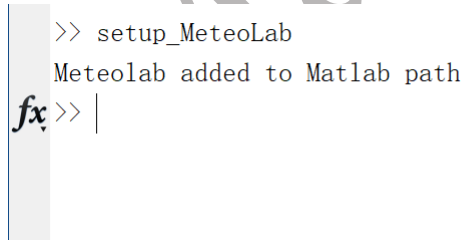
1. 需要先将当前文件夹设置为工具箱安装路径



2. 在命令行输入 setup_MeteoLab



3. 出现 Meteolab added to Matlab path, 即为启动成功



三、函数使用说明

2.1 基本常数

2.2 基础物理量

科氏参数

$$f = 2\Omega \sin\varphi$$

$$\beta = 2\Omega \cos\varphi$$

`function [f,b]=coriolis_parameter(lat,lon)`

输入纬度 `lat(1*n)`、经度 `lon(1*m)`

输出科氏参数 `f(n*m)` 及 β 项(科氏参数随纬度的导数) `b(n*m)`

经纬度网格距

`function [dx,dy]=latlon2delta(lat,lon)`

输入纬度 `lat(1*n)`、经度 `lon(1*m)`

输出纬向、经向网格距(m)(目前仅适用于等经纬度和墨卡托投影网格) `dx,dy(n*m)`

空气密度

$$\rho = \frac{p}{RT}$$

`function rho=density(T,pressure)`

输入温度 `T(vertical,lat,lon)`, 气压(Pa) `pressure(vertical)`

输出空气密度 `rho(vertical,lat,lon)`

散度(二维)

`function dv=divergence_2d(U,V,lat,dx,dy)`

输入经向风、纬向风 `U,V(n*m)`, 纬度 `lat(1*n)`, 纬向、经向网格距 `dx,dy(n*m)`

输出二维散度 `dv(n,m)`

垂直相对涡度

`function rv=vorticity_2d(U,V,dx,dy,lat)`

绝对涡度

`function Ca = absolute_vorticity(U,V,lat,dx,dy)`

水平平流项

$$adv = \vec{V} \cdot \nabla F$$

`function adv=advection_2d(F,U,V,dx,dy)`

输入物理量 `F(n*m)`, 经向风、纬向风 `U,V(n*m)`, 纬向、经向网格距 `dx,dy(n*m)`

输出二维平流项 `adv(n,m)`

2.3 差分/梯度/平滑

水平梯度

`function [gradx,grady]=gradient_2d(F,dx,dy)`

输入物理量 `F(n*m)`, 纬向、经向网格距 `dx,dy(n*m)`

输出等压面/等位势面上的水平梯度 `gradx,grady(n,m)`

垂直梯度

`function [gradp]=gradient_vert(F,vertical)`

输入物理量 `F(vertical,n,m)`、垂直方向的坐标(气压或高度) `vertical(vertical)`

输出物理量的垂直梯度 `gradp(vertical,n,m)`

拉普拉斯项(二维)

`function Lap=Laplacian_2d(F,dx,dy)`

输入物理量 `F(n*m)`, 纬向、经向网格距 `dx,dy(n*m)`

输出等压面/等位势面上的拉普拉斯项 `Lap(n,m)`

拉普拉斯项(垂直)

`function Lap_vert=Laplacian_vert(F,vertical)`

输入物理量 `F(vertical,n,m)`、垂直方向的坐标(气压或高度) `vertical(vertical)`

输出物理量垂直方向的拉普拉斯项 `Lap_vert(vertical,n,m)`

五点平滑

```
function F1=smth5(F,s)
```

输入物理量 `F(n*m)`、平滑系数 `s`

输出五点平滑后的数据 `F1(n,m)`

九点平滑

```
function F1=smth9(F,s)
```

输入物理量 `F(n*m)`、平滑系数 `s`

输出九点平滑后的数据 `F1(n,m)`

2.4 动力相关量

地转风

```
function [Ug,Vg]=geostrophic_wind(H,dx,dy,f)
```

输入位势高度(位势米) `H(n*m)`、纬向、经向网格距 `dx,dy(n*m)`、科氏参数 `f(n*m)`

输出地转风场 `Ug,Vg(n,m)`

Q 矢量

```
function [Qx,Qy]=qvector_isobaric(T,Ug,Vg,pressure,dx,dy)
```

输入温度 (K) `T(n*m)`、地转风场 `Ug,Vg(n*m)`、气压 (Pa) `pressure`、纬向、经向网格距 `dx,dy(n*m)`

输出 Q 矢量 `Q(n,m)`

正压模式下的位势涡度

```
function pv = potential_vorticity_barotropic(U,V,H,lat,dx,dy)
```

Ertel 位涡

```
function pv = Ertel_potential_vorticity(theta,U,V,dx,dy,lat,pressure)
```

湿位涡

```
function [MPV,MPV1,MPV2] = ...  
    moist_potential_vorticity(RH,T,U,V,lat,dx,dy,pressure)
```

2.5 热力相关量

位温

```
function theta=potential_temperature(T,pressure)
```

计算三维空间上的位温

输入温度 (K) `T(vertical,n,m)`、气压 (Pa) `pressure(vertical)`

输出位温 `theta(vertical,n,m)`

相当位温

```
function theta_e=equivalent_potential_temperature(RH,T,pressure)
```

计算三维空间上的相当位温

输入相对湿度 (单位不是%) `RH(vertical,n,m)`、温度 (K) `T(vertical,n,m)`、气压 (Pa) `pressure(vertical)`

输出相当位温 `theta_e(vertical,n,m)`

干绝热递减率

```
function gamma_d = dry_lapse(T,pressure)
```

湿绝热递减率

```
function gamma_s = moist_lapse(T,pressure)
```

非绝热加热率

```
function H = diabatic_heating(T,Omega,pressure,RH)
```

锋生函数

```
function F = frontogenesis(theta,U,V,dx,dy)
```

2.6 水汽相关量

露点温度

```
function Td=Dewpoint(RH,T)
```

输入相对湿度（单位不是%）RH、温度（K）T

输出露点温度 Td

抬升凝结温度

```
function Tlcl=T_lcl(T,Td)
```

计算抬升凝结高度处（LCL）的温度

输入温度（K）T、露点温度 Td

输出抬升凝结温度 Tlcl

水汽通量/散度

```
function [qu,qv,qd] = vapor_flux(Q,U,V,dx,dy,lat)
```

饱和水汽压

```
function es = saturation_vapor_pressure(T)
```

饱和混合比

```
function rs = saturation_mixing_ratio(T,pressure)
```

2.7 大气稳定性

静力稳定度

```
function [ss]=static_stability(T,pressure)
```

输入温度（K）T(vertical,n,m)、气压(Pa)pressure(vertical)

输出静力稳定度 ss(vertical,n,m)

四、高级功能

1. Shuman-Shapiro 滤波

```
function [synoptic_scale,meso_scale]=  
shuman_shapiro_filter(F,s,option)
```

2. 涡度收支诊断

```
function [H_adv,V_adv,Tilt,Dive]=  
vorticity_equation(U,V,W,vertical,dx,dy,lat)
```

3. 台风诊断

二维直角坐标插值到二维极坐标系

```
function [F,theta,r,lat_n,lon_n] = hurricane_cart2pol ...  
(F0,ctr_lat,ctr_lon,lat,lon,Radius,Nr,Ntheta,option)
```

%% 功能：以涡旋中心为原点，将二维直角坐标插值到极坐标

%使用方法：

%输入变量：F0：气象要素，ctr_lat,ctr_lon：涡旋中心经纬度

%lat,lon：一维经纬度向量，R：需要插值的极坐标半径范围（km 或度）

%Nr：极坐标径向格点个数，Ntheta：方位格点个数

%!!

%!!! option: 径向方向的单位是距离还是度, 是距离则输入 0, 度则输入 1

%!!

%输出变量: F: 插值后的气象要素, theta: 方位坐标, r: 径向坐标

%lat_n, lon_n: 插值后气象要素的经纬度坐标

三维直角坐标插值到柱坐标系

```
function [F,theta,r,lat_n,lon_n] = hurricane_cart2cyl ...
```

```
(F0,ctr_lat,ctr_lon,lat,lon,vertical,Radius,Nr,Ntheta,option)
```

% 功能: 以涡旋中心为原点, 将三维直角坐标插值到柱坐标

%使用方法:

%输入变量: F0: 气象要素, ctr_lat,ctr_lon: 涡旋中心经纬度

%lat,lon: 一维经纬度向量, vertical: 一维垂直坐标 (气压或高度)

%R: 需要插值的极坐标半径范围 (km 或度)

%Nr: 极坐标径向格点个数, Ntheta: 方位格点个数

%!!

%!!! option: 径向方向的单位是距离还是度, 是距离则输入 0, 度则输入 1

%!!

%输出变量: F: 插值后的气象要素, theta: 方位坐标, r: 径向坐标

%lat_n, lon_n: 插值后气象要素的经纬度坐标