PyPASS: Python-based Performance Analysis Supporting System

Byeong-Uk Kim and Harvey E. Jeffries

Department of Environmental Sciences and Engineering

University of North Carolina

Why a new tool?

- High quality information is important.
  - ‘Information’ means any communication or representation of knowledge such as facts or data, in any medium or form, including *textual, numerical, graphic, cartographic, narrative, or audiovisual forms*. - OMB’s definition of ‘information’
- Effective tools in applying science are important.
  - “We must therefore be more efficient about the way we apply science through modelling, so as to leave sufficient time to do science!” – Argent, 2004
- Model performance evaluation plays a key role in using modeling results for decision making processes.
- **PyPASS can provide high quality information related to model performance evaluation with high efficiency.**

Goals of PyPASS development

- Efficient and rapid information generation
  - Visualizing and summarizing observation and prediction with predefined graph/document formats
- More context-rich information
  - Processing virtually all observed/modeled variables (e.g meteorological/chemical signal)
- Open/free (or low cost) software
  - Supporting performance evaluation of air quality modeling studies with little extra resources including licensing fee

Operation of PyPASS

- Command-line driven package such as ‘ls’
- Platform/model supported
  - CAMx/CMAQ – CAMxSubset/CMAQExtract utilities are mandatory
  - Windows XP/Linux/MacOSX
- Example of PyPASS execution:
  - `python %PYPASS%\MakeTSSSPlots.py -f opts\NONO2O3_UHTCEQSIPBASE_OneDay.opt -b2000-08-22 -e2000-08-23 HGBBPAMonData000818_000906.h5 b5b.sipcase.uh-tceq.h5`

    Time window — Monitor data — Model dataset

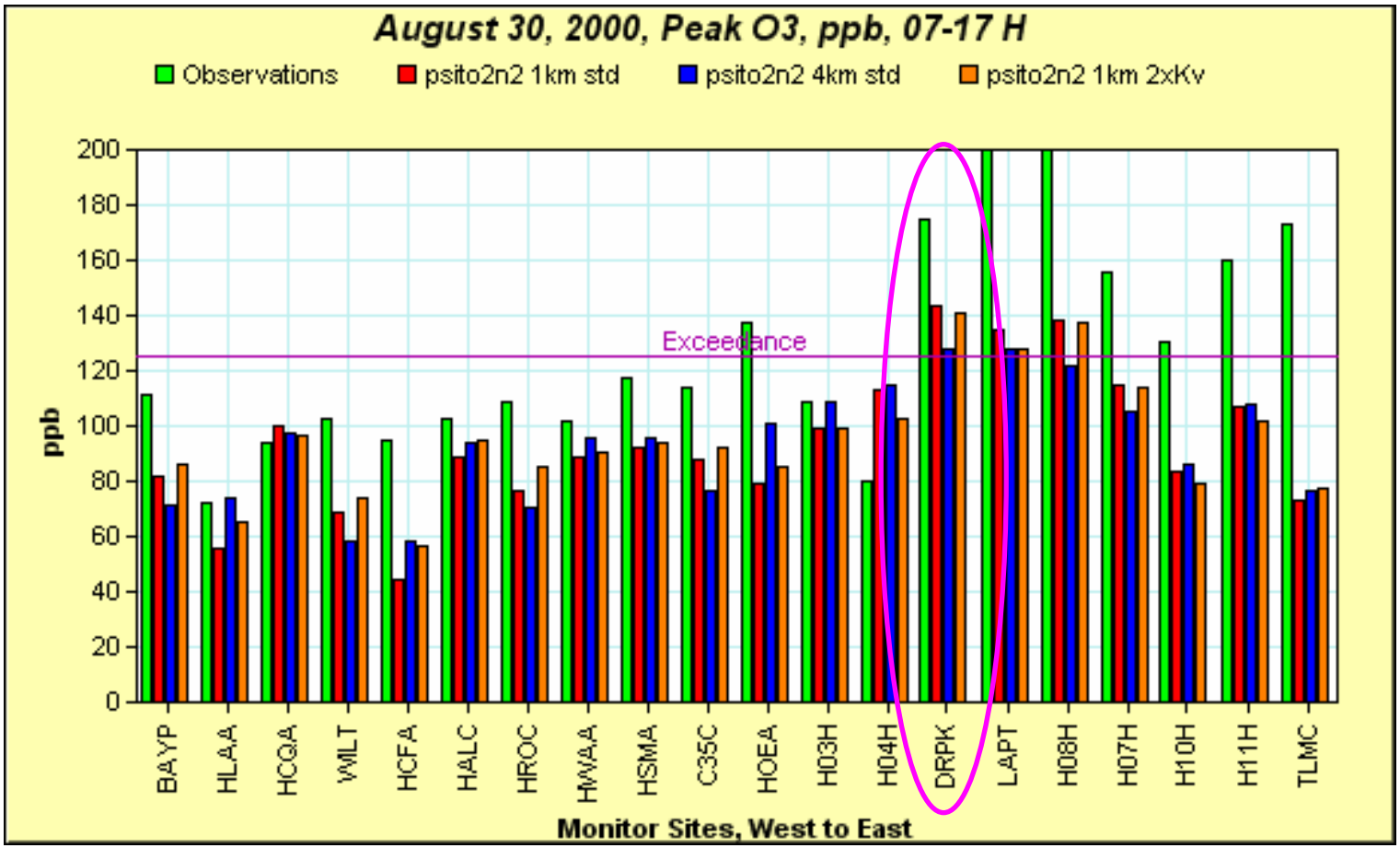
    options

```
-i hdf5
-o output\plotout\BP_SS_TS\UHTCEQSIPBASE\ts\
-p OneDay_pph -y CStd160 -l -v 125.0 -n 3S2S_4k
-s NO  -c B -g Circle -s NO2 -c G -g Diamond -s O3  -c R -g Square
-m b5b_psito2n2_4km_v6_0822_0831 -d SDash
-m b5b_psito2n2_4km_cmaq_0823_0831 -d FDot
-m b5b_q20_4km_cmaq_0823_0831 -d Dot
```

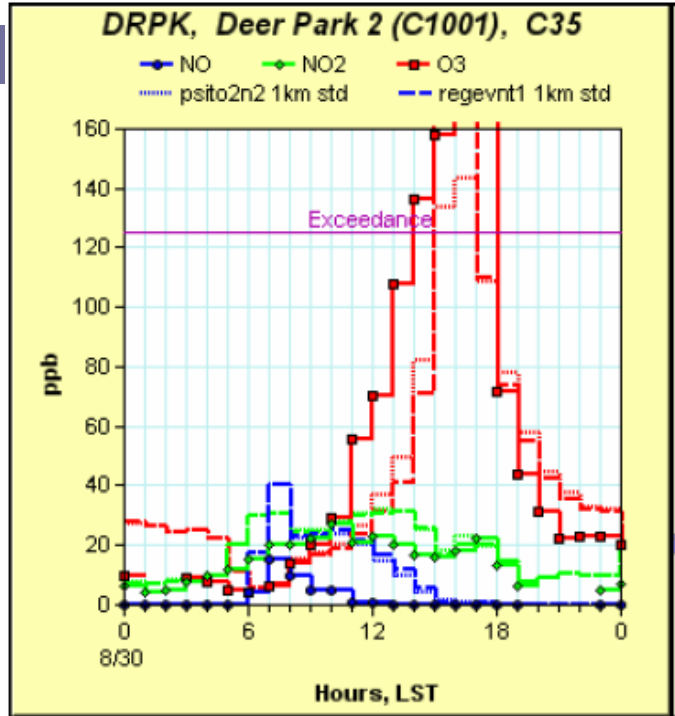
Examples of PyPASS outputs

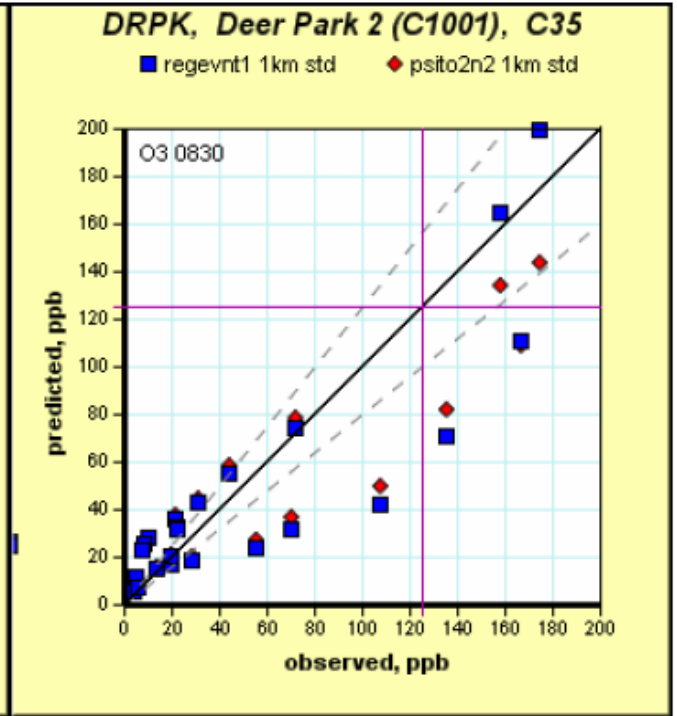
- Bar charts
- Scatter plots
- Time series plots
- Tile plots (XY/XZ/YZ orientation)
  - Optional features: roads, monitors, wind vectors and more on cell/real coordinates for axes
- Statistics
- Reports
  - MS-WORD/HTML/PDF/TEX

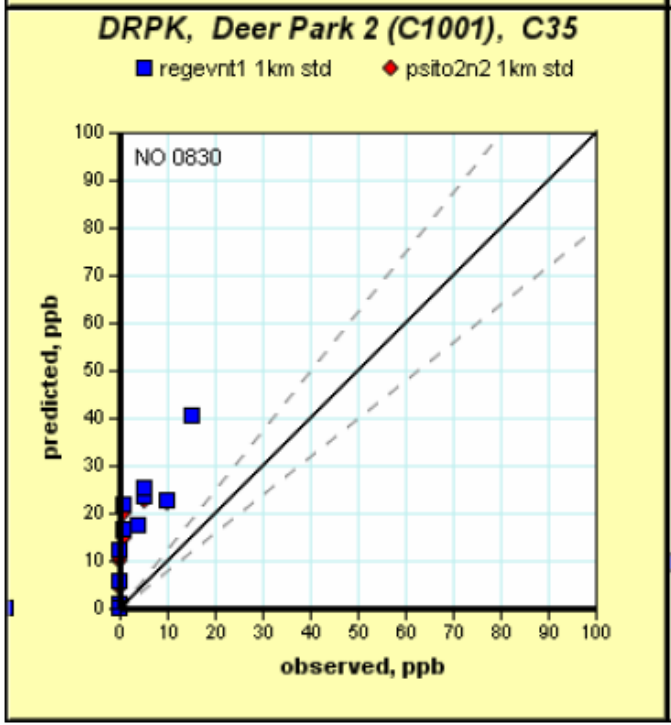
Bar chart

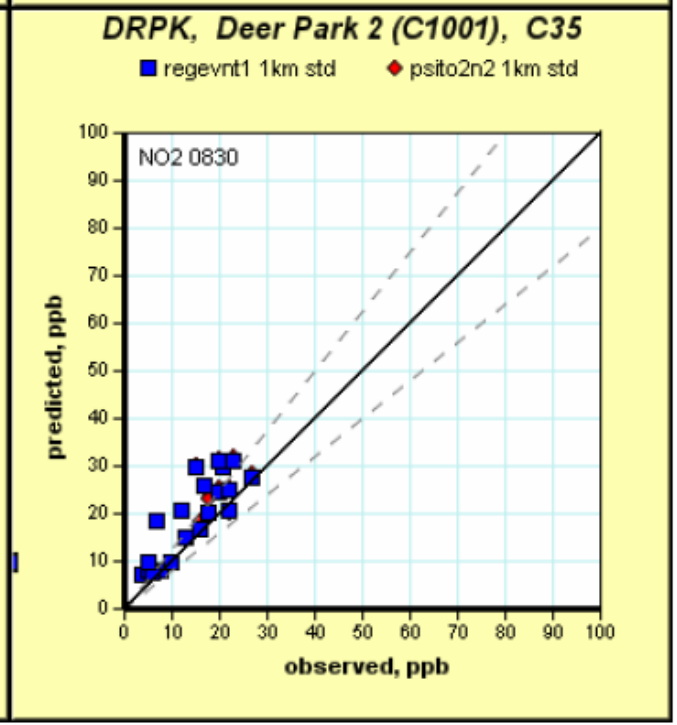

August 30, 2000, Peak O3, ppb, 07-17 H
Observations
psito2n2 1km std
psito2n2 4km std
psito2n2 1km 2xKv
ppb
200
180
160
140
120
100
80
60
40
20
0
Exceedance
BAYP
HLAA
HCQA
WILT
HCFA
HALC
HROC
HWAA
HSMA
C35C
HOEA
H03H
H04H
DRPK
LAPT
H08H
H07H
H10H
H11H
TLMC
Monitor Sites, West to East

Time series and scatter plots for chemical species

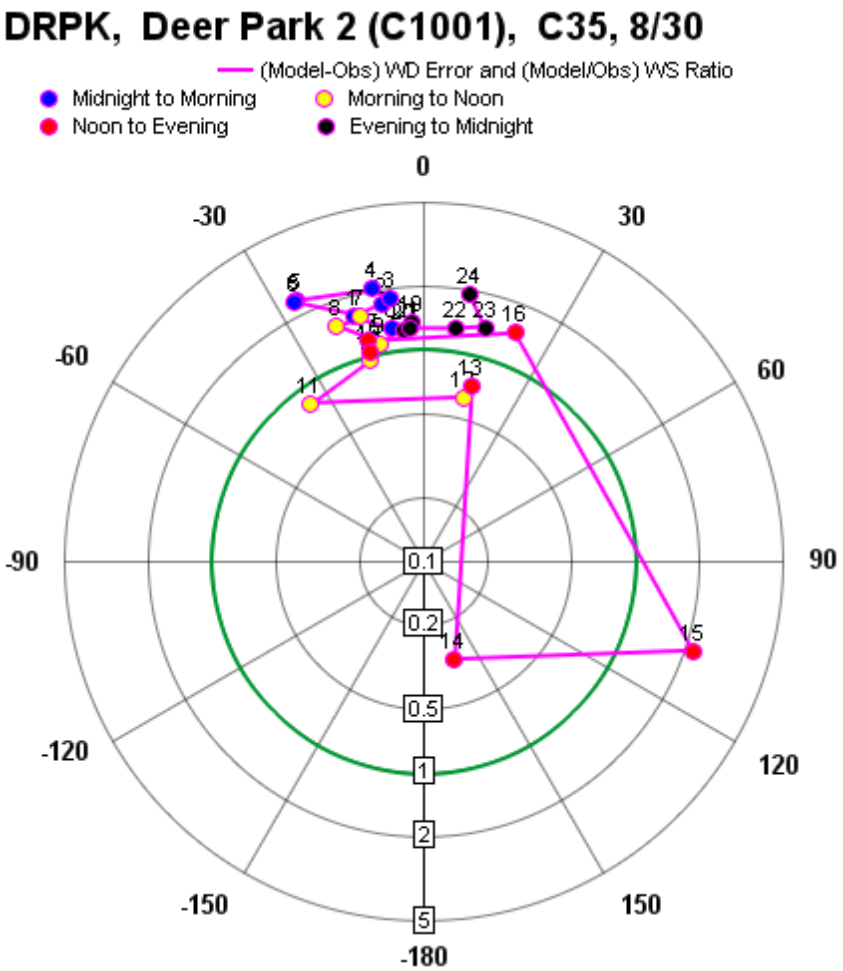


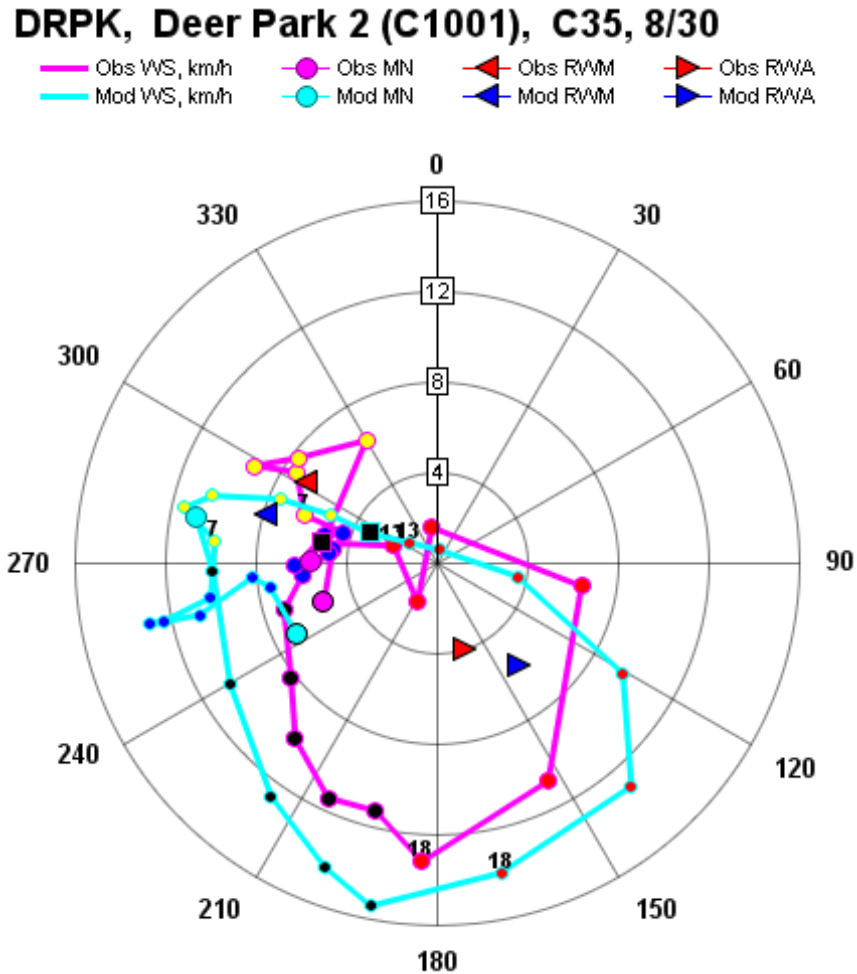




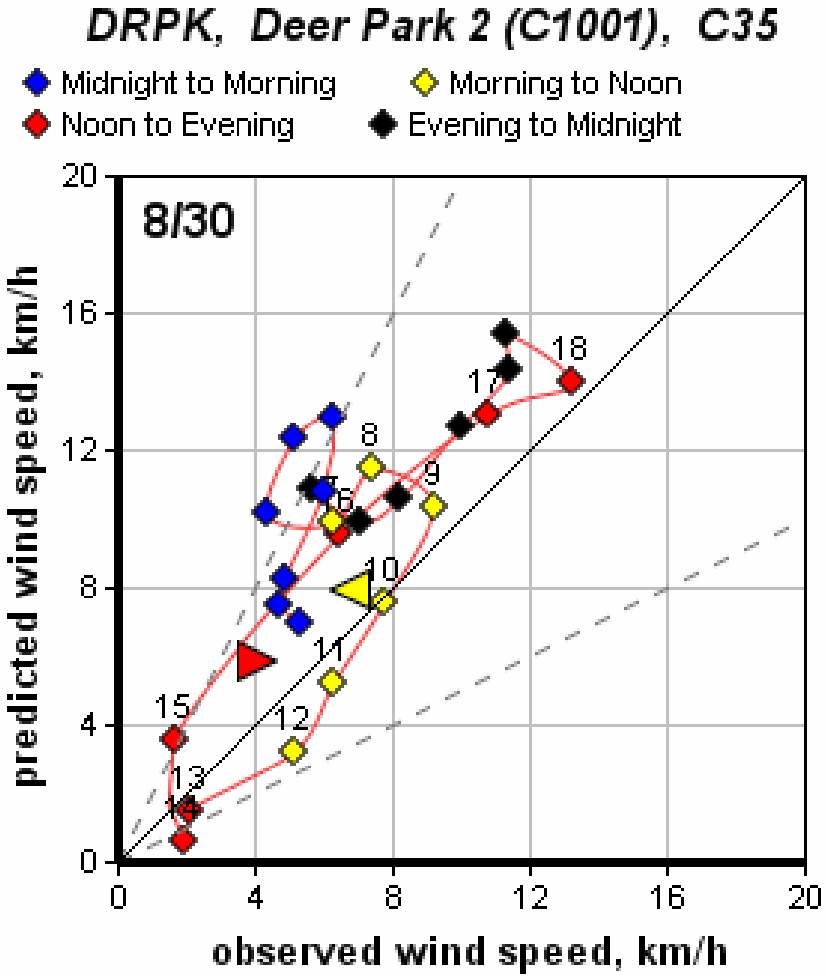


Hodogram: Time series for surface winds

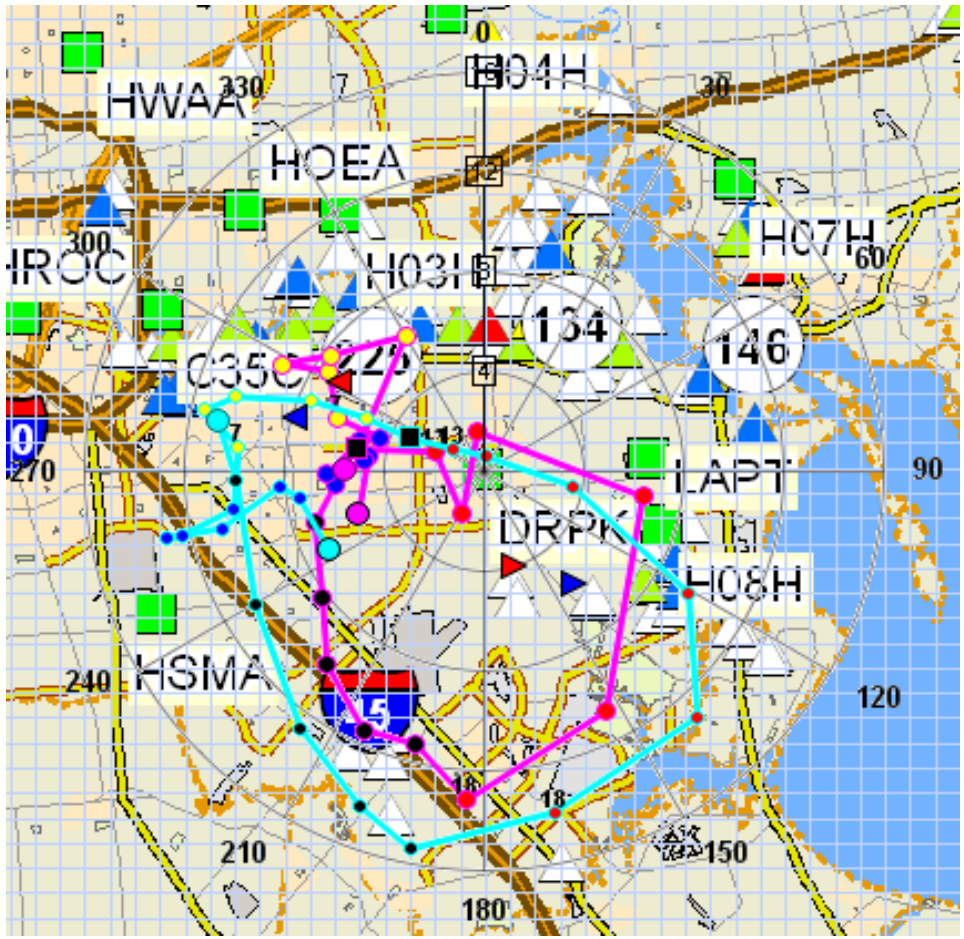




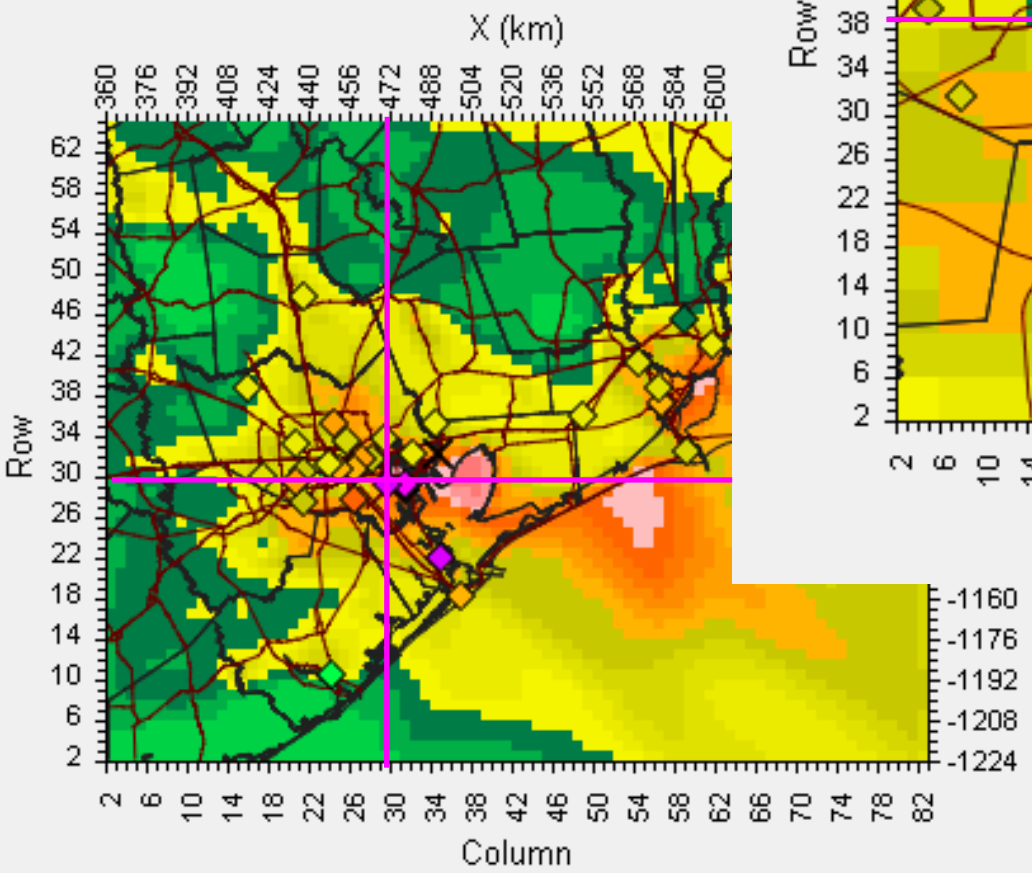
Scatter plot for surface winds

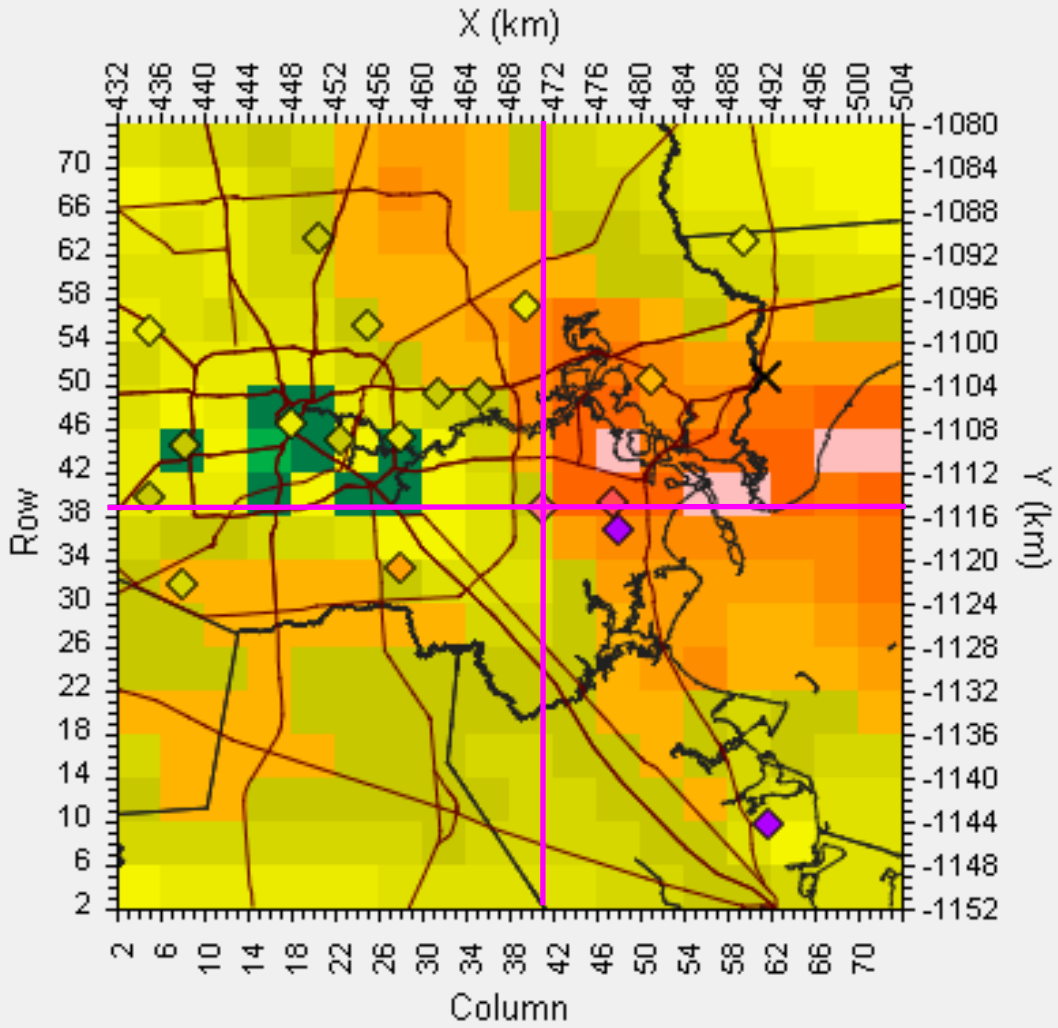


Hodogram over GIS map

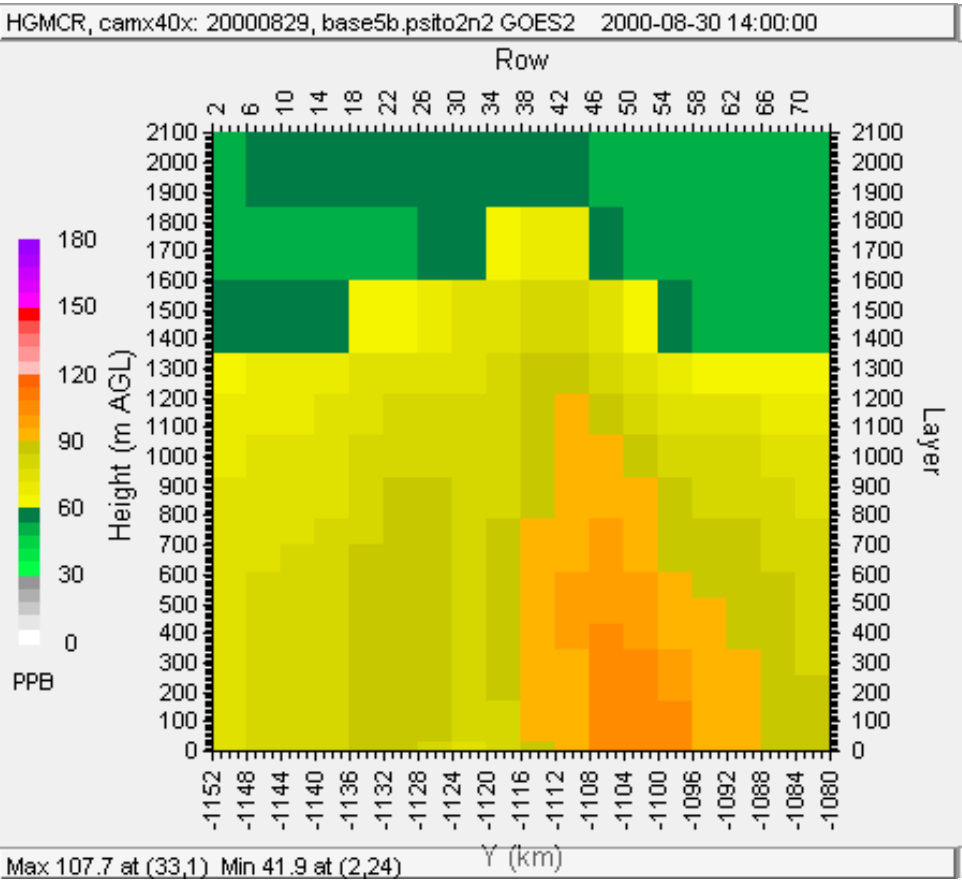


Tile plot on different grids with same data

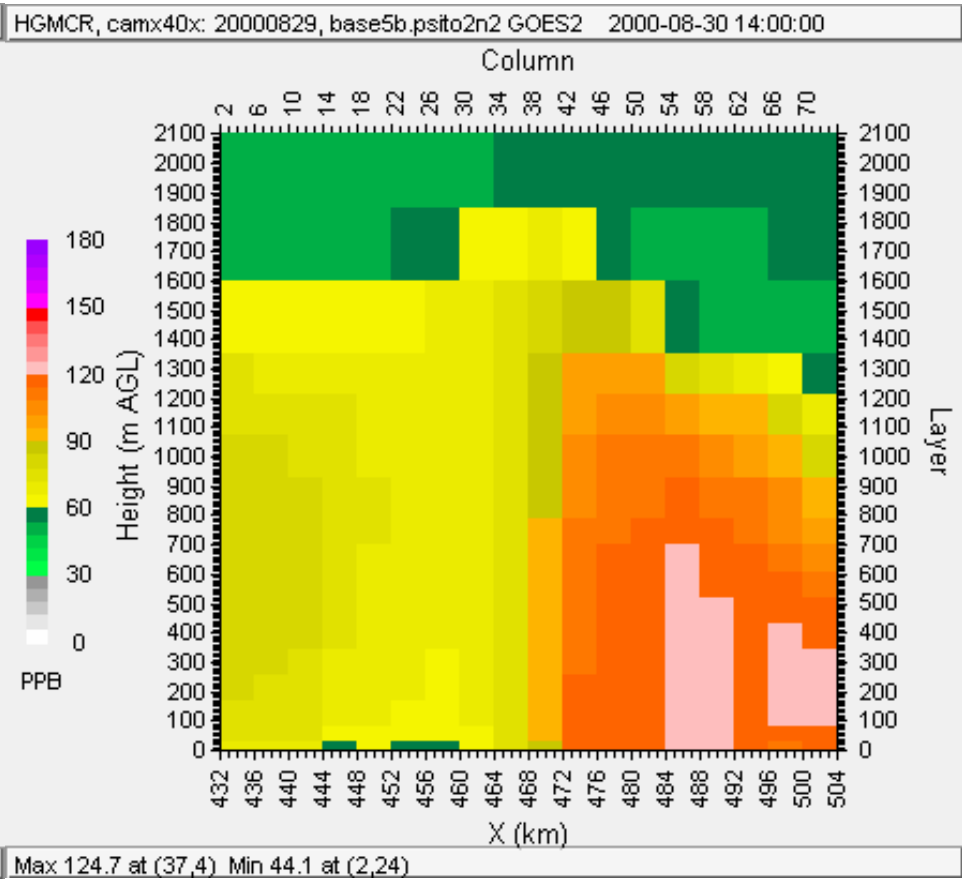




Vertical tile plots

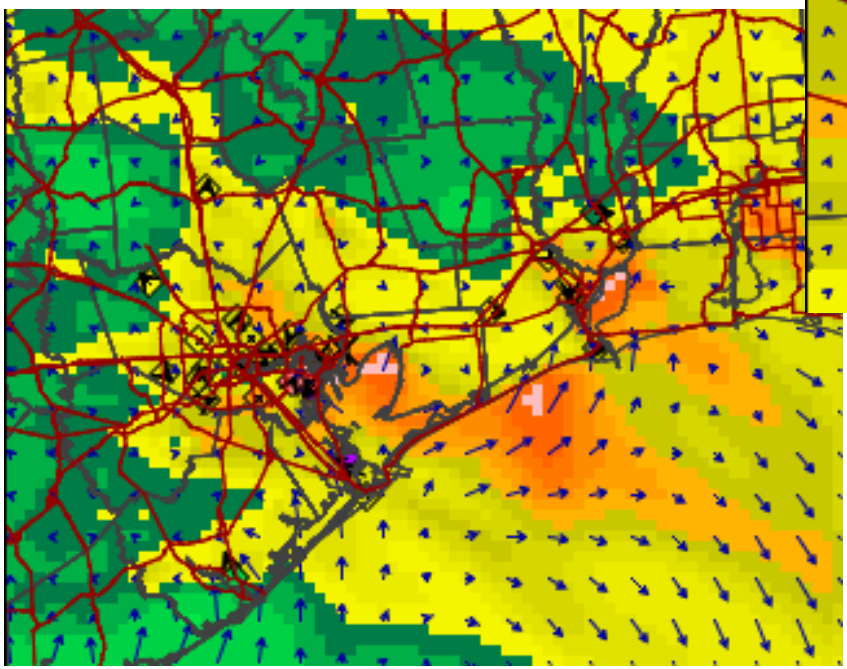


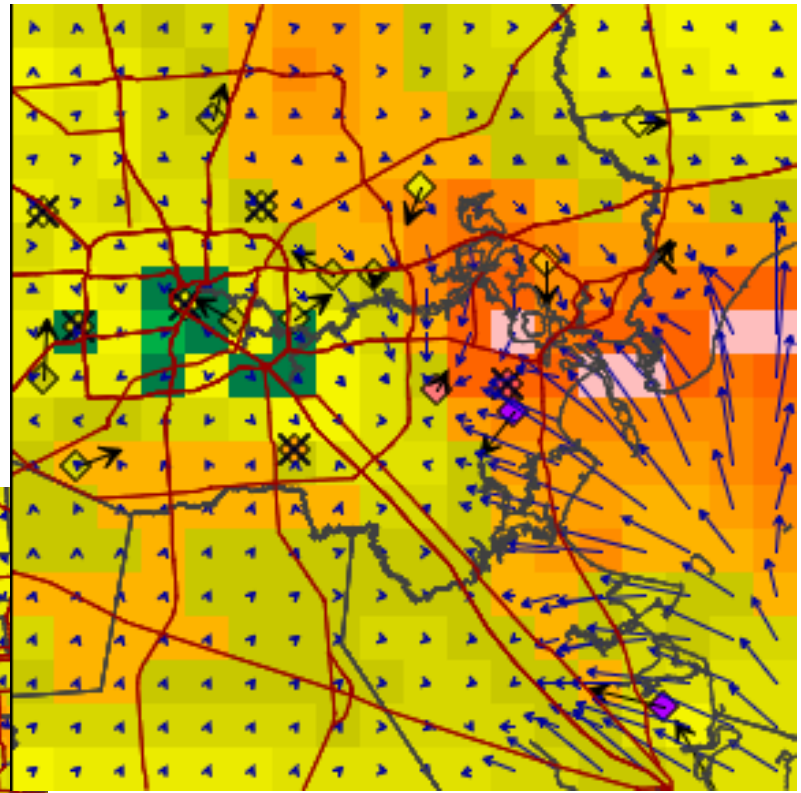
Column 29

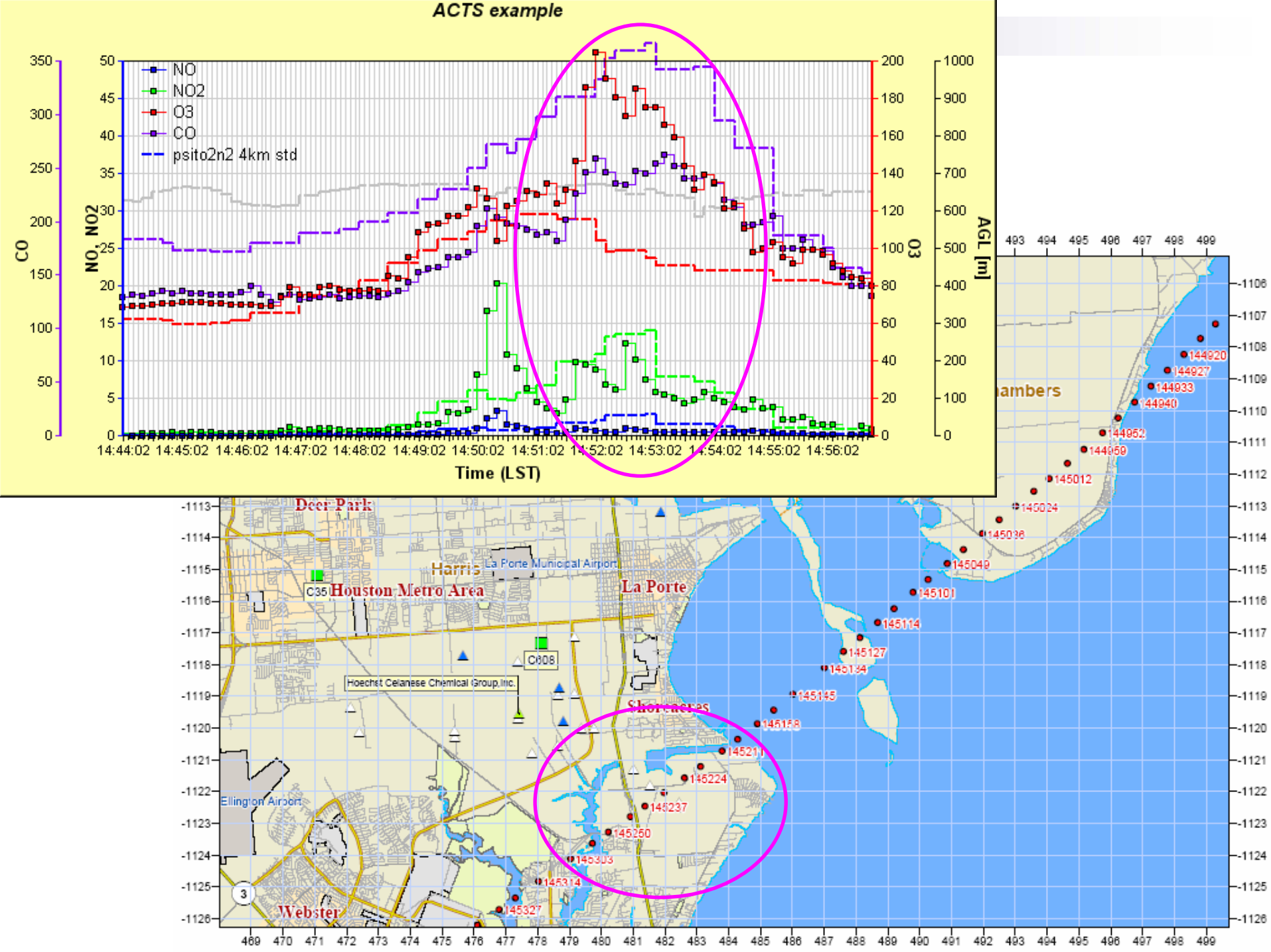


Row 29

Tile plot
with wind field






ACTS example
NO
NO2
O3
CO
psito2n2 4km std
CO
NO, NO2
O3
AGL [m]
Time (LST)
14:44:02 14:45:02 14:46:02 14:47:02 14:48:02 14:49:02 14:50:02 14:51:02 14:52:02 14:53:02 14:54:02 14:55:02 14:56:02
Deer Park
Harris
La Porte Municipal Airport
C35
Houston Metro Area
La Porte
C808
Hoechst Celanese Chemical Group,Inc.
Shoreacres
Ellington Airport
Webster
3
144920
144927
144933
144940
144952
144959
145012
145024
145036
145049
145101
145114
145127
145134
145145
145158
145211
145224
145237
145250
145303
145314
145327

Example of statistical measures

Mean Normalized Bias (*MNB*)

$$MNB = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(C_p(x_i,t) - C_o(x_i,t)\right)}{C_o(x_i,t)}, \quad t = 1, 24$$

Mean Normalized Gross Error (*MNGE*)

$$MGE = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|C_p(x_i,t) - C_o(x_i,t)\right|}{C_o(x_i,t)}, \quad t = 1, 24$$

Unpaired Peak Prediction Accuracy (*UPPA*)

$$UPPA = \frac{C_p(x,t)_{\max} - C_o(x',t')_{\max}}{C_o(x',t')_{\max}} \times 100\%$$

Modified Index of Agreement, d_1

$$d_1 = 1.0 - \frac{\sum_{i=1}^{N}\left|O_i - P_i\right|}{\sum_{i=1}^{N}\left(\left|P_i - \overline{O}\right| + \left|O_i - \overline{O}\right|\right)}$$

Modified Coefficient of Efficiency, E_1

$$E_1 = 1.0 - \frac{\sum_{i=1}^{N}\left|O_i - P_i\right|}{\sum_{i=1}^{N}\left|O_i - \overline{O}\right|}$$

d1 and E1 from Legates and MaCabe Jr., 1999

```
# Mean Normalized Bias
MNB = ((mod-obs)/obs).sum()/N
# Mean Normalzied Gross Error
MNGE = (abs(mod-obs)/obs).sum()/N
# Unpaired peak accuracy
UPPA = (mod.max()-obs.max())/obs.max() * 100. # output is %
# Modified Index of Agreement, d1
avgobs = obs.sum()/N
d1 = 1.0 - (abs(obs-mod)).sum()/(abs(mod-avgobs)+abs(obs-avgobs)).sum()
# Modified Coefficient of Efficiency, e1
E1 = 1.0 - (abs(obs-mod)).sum()/(abs(obs-avgobs)).sum()
```

SITE	MNB	MNGE	UPPA	d1	E1	Contingency Table				# of valid data
BAYP	-0.063	0.272	-25.6	0.772	0.589	0	0	0	20	20
HLAA	0.297	0.350	2.4	0.846	0.695	0	0	0	22	22
HCQA	0.391	0.391	4.1	0.800	0.581	0	0	0	18	18
WILT	-0.436	0.438	-35.3	0.612	0.249	0	0	0	24	24
HCFA	-0.058	0.372	-35.5	0.745	0.547	0	0	0	20	20
HALC	0.309	0.344	-9.0	0.832	0.693	0	0	0	17	17
HROC	0.159	0.596	-25.2	0.683	0.430	0	0	0	20	20
HWAA	0.445	0.457	-6.4	0.784	0.594	0	0	0	14	14
HSMA	0.509	0.568	-18.0	0.746	0.552	0	0	0	18	18
C35C	-0.310	0.459	-29.7	0.702	0.473	0	0	0	12	12
HOEA	-0.141	0.353	-26.5	0.851	0.722	0	0	1	16	17
H03H	-0.235	0.396	-0.5	0.887	0.753	0	0	0	21	21
H04H	0.286	0.496	9.9	0.710	0.288	0	0	0	18	18
DRPK	0.111	0.408	-26.8	0.768	0.595	1	0	3	17	21
LAPT	-0.172	0.239	-36.1	0.802	0.646	2	0	3	18	23
H08H	-0.456	0.460	-39.0	0.716	0.476	0	0	5	16	21
H07H	-0.201	0.410	-32.2	0.778	0.552	0	0	1	17	18
H10H	0.228	0.414	-32.2	0.771	0.591	0	0	2	16	18
H11H	-0.316	0.420	-32.5	0.777	0.607	0	0	2	6	8
TLMC	0.527	0.758	-55.7	0.576	0.396	0	0	3	16	19
HNWA	0.617	0.704	-6.3	0.535	0.206	0	0	0	18	18
SHWH	0.510	0.527	-9.2	0.798	0.618	0	0	0	18	18
CONR	1.157	1.157	4.6	0.592	0.159	0	0	0	22	22
CLTA	0.400	0.670	69.7	0.560	-0.357	0	0	0	21	21
GALC	0.539	0.564	-12.0	0.617	0.277	0	0	0	22	22
JEFC	0.375	0.450	-34.7	0.649	0.427	0	0	1	17	18
BMTC	0.098	0.364	-32.4	0.735	0.500	0	0	1	21	22
S43S	0.293	0.382	-33.0	0.699	0.460	0	0	2	17	19
PAWC	0.231	0.356	-39.6	0.789	0.645	0	0	2	20	22
S42S	0.901	0.961	10.1	0.415	-0.236	0	0	0	18	18
S40S	0.579	0.722	-37.2	0.682	0.493	0	0	4	16	20
WORA	0.300	0.489	-24.1	0.521	0.202	0	0	1	17	18

Performance of PyPASS

- Evaluation condition
  - H/W: P4 3.2 GHz/2GB RAM/3Dlabs Wildcat VP990 Pro
  - O/S: Windows XP (Service Pack 2)
  - Test period: three species on one day (one plot for each of time series plot or scatter plot, 24 hours plots for tile plots)
- Time
  - Wind field overlaid tile plots in HGA_4km outputs
    - HGA_4km (83*65): ~ 2 minutes
    - HGA_1km (74*74): ~ 30 seconds
  - Time series plots: 2 seconds
  - Scatter plots: 3 seconds
- Storage
  - BIN file only holds the necessary dataset from the original CAMx/CMAQ ouput
    - TX HGMCR modeling case: 451 kB holding data of 15 species on 32 monitors for 16days
  - PyTable supports compression
    - Approximately ~40% saving compared with uncompressed file (i.e. BIN file)