Statistical approaches for air pollution prediction

Giorgio Corani
IDSIA, Switzerland
giorgio@idsia.ch

Méthodes Statistiques et Pollution
20 June 2008, INSA de Rouen

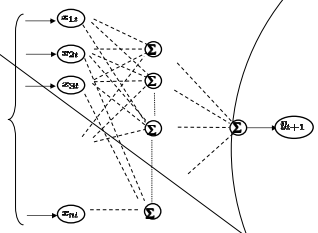
Outline

- Statistical approaches:
  - neural networks
  - pruned neural networks
  - **lazy learning**
- Air pollution prediction in the Milan area:
  - ozone prediction;
  - Pm10 prediction.

Air pollution prediction with statistical algorithms

- Air pollution is a complex phenomenon, for which non-linear modelling approaches are recommended.
- Statistical models do not provide an explanation of the phenomenon, but can deliver accurate predictions.
- Predictions are punctual: they are issued for a specific point, i.e., a monitoring station for which historical records are available.
- For a review, see "A rigorous inter-comparison of ground-level ozone predictions", Schlink et al., 2003

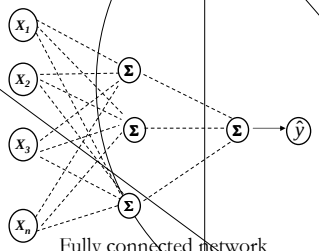
Feed-forward neural networks (FFNNs)



- Universal approximator
- High accuracy in non-linear modelling
- Large number of applications in air pollution prediction

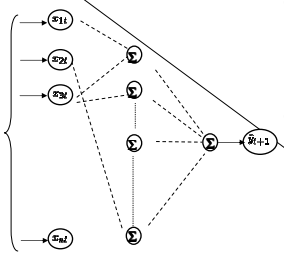
FFNNs weaknesses

- Lack of an analytical model selection approach.
- The optimal architecture and its parameters have to be found by trial and error, which is *time-consuming*.
- Difficult to assess the inputs relevance (air pollution has a huge number of possible inputs: which ones do really matter?)



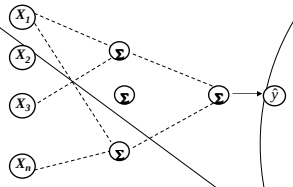
Fully connected network

Pruned neural networks (PNNs)



- Pruning algorithms start from a FFNN and remove unnecessary parameters.
- PNNs are non-linear and parameters-parsimonious.
- *Free toolbox for matlab:*
  http://www.iau.dtu.dk/research/control/nnsysid.html.

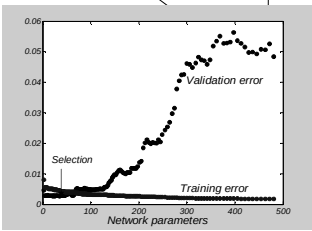
Pruned neural networks (II)



- The architecture is designed almost automatically by pruning algorithms: reduced need of guesswork, but several restarts are needed.
- The pruned architecture might have disconnected all the links referring to a certain input, which is hence recognizes as irrelevant
- Better readability of the model structure compared to FFNNs

Pruned neural networks (III)

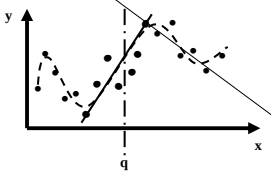
- Split the data into three folds: training, validation, testing
- Pruning uses both training and validation set to determine the architecture
- Eventually, the performance has to be assessed on an independent data set, i.e., the testing set



A local linear approach: lazy learning (LL)

- *Query point*: an instance of measures of input variables, in corrispondence of which the target variable has to be predicted.
- E.g.: if we have to predict c_{t+1} using a_t and b_t, the query-point is $[a_t, b_t]$
- Lazy learning provides a linear approximation of the true unknown non-linear function in the neighborhood of the query-point.
- The algorithm does not learn until the query-point is provided: *lazy learning*!

Lazy learning (II)



REMARKS

- Data have to be normalized before computing the distances
- Sophisticated techniques to tune *k* query-by-query (Bontempi & Birattari, 1999)

LL procedure on a given query-point:

- Rank the historical query-points according to the distance from the provided query-point;
- Select the *k* closest samples;
- Fit a local regressors on these *k* samples and returns the prediction

Lazy learning (III)

- For each query-point, a linear local regressor is identified, used to issue the prediction and then discarded.
- *Memory based* : the training set has to be kept in memory to answer the queries
- *Easy to update*: it is enough to add the new instances to the training set, without any re-training
- LL gives linearity a chance: fast and exact algorithm from linear statistics can be imported (parameters identification, model selection)

Lazy learning references

- Query by query bandwidth tuning makes LL very powerful.
- Fast implementation available for Matlab and R (see: Bontempi & Birattari, NIPS 1999): https://iridia.ulb.ac.be/~lazy/
- This implementation tunes *k* query-by-query and also implements combination of local regressor; in this way, it delivers high accuracy
- Application to air pollution: Corani, 2005

Summary

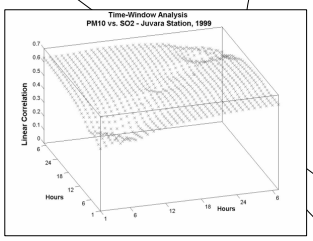
	FFNNs	PNNs	Lazy Learning
Non-linear	**yes**	**yes**	**yes**
High accuracy	**yes**	**yes**	**yes**
Training	***slow***	***slow***	**quick**
Prediction	**quick**	**quick**	**quick** (thanksB&B)
Model readability	**no**	**moderate**	**yes**
Easy update	**no**	**no**	**yes**

Input selection

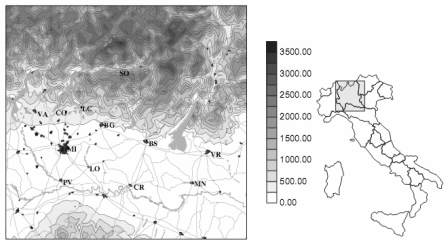
- Correlation-based: look for a set of features which maximizes the correlation with the target variables, while minimizing the cross-correlation between the inputs.
- Alternative approaches could be more sophisticated, yet correlation works and is fast.

Input aggregation: from hourly to daily

- Inputs variables are grouped to daily values, by averaging over a hourly range (0-1 a.m, 0-2, 0-3, ...22-23).
- The range which maximizes correlation is finally chosen.



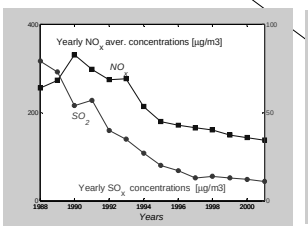
Milan case study

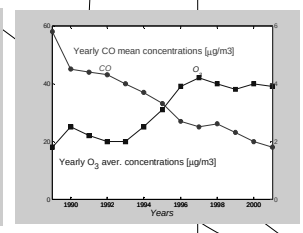


Location and orography surrounding Milan (m a.s.l.)
Strongly populated and industrialized area.

Air pollutants trends

- Over the period 1989-2001, significant reduction of the yearly average of SO2, NOx, CO, TSP (-90%, -50%, -65%, -60%).
- Also benzene has been reduced below the law limts.
- An important concern: O3 increasing from the early 90's.





Ozone

- Adverse effects for both humans and agricultural crops
- Ozone formation takes place at high temperatures (30 C) requiring the presence of "precursors" (it is a secondary polluttant):
  - NOx (mainly due to road transports)
  - VOC (volatile organic compound) mainly emitted by solvent use.
- The Milan area is *VOC-limited*

Pm10

- Perhaps, the biggest concern is PM10. Its yearly average has been between 50 and 55µg/m^3 since the beginning of the monitoring (1998).
- The 50 µg/m^3 threshold on daily average is exceeded about 130 days every year (against 35 recommended by EU).

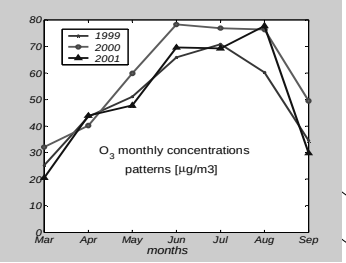
Ozone application

The problem

- Human health target: 110 µg/m3 on the maximum 8-hour moving average
- The problem: to predict at 9 a.m. the max 8-hours average for the current day
- Data set: 1999 -2001, for a monitoring station not directly exposed to the traffic
- Besides ozone, many meteorological and air quality parameters are recorded by the same station

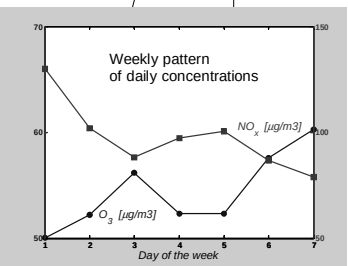
Ozone periodicities

- Because of its dependence on solar radiation and temperature, ozone reaches its maximum in summer, while being negligible in winter
- Analysis restricted to the period April-September



Ozone periodicities (II)

- Important weekly periodicity, due to the cycles of anthropic activities and emissions:
- NOx (traffic tracker) strongly reduces during weekends..
- while O3 on the other hand increases (weekend effect)



Input selection

- For each input variable, the 24 hourly values should be grouped into a single daily value (e.g., averaging over a time window)
- The inputs are selected via cross-correlation analysis

Variable	Aggregation	Variable	Aggregation
O_3	$\max[\mu_8(t-1)]$	Temp.	$\mu[6_t-9_t]$
O_3	9_t	Hum.	$\mu[6_t-9_t]$
NO	$\mu[6_t-9_t]$	Global Sol. Rad.	$\mu[8_t-9_t]$
NO_2	μ_{t-1}	W. speed	μ_{t-1}
CO	$\mu[6_t-9_t]$	Rain	μ_{t-1}
Press.	μ_{t-1}	Stab. class.	μ_{t-1}

Table 3.1: Input variables chosen for ozone prediction. $\mu(t)$ denotes the average operator.

Data pre-processing

- De-seasonalization: to remove the periodic components from the data, to easy the learning task.
- Simple yet effective: *weekly-based standardization*

  Different mean and variances are used to standardize the data of air pollution variables, depending on whether the day falls during the week or during the week-end

Performance indicators (I)

- MAE (mean absolute error) – the lower, the better
- MBE (mean bias error, or simply mean error) – the closer to 0, the better
- ρ (correlation) – the closer to 1, the better
- d (index of agreement) – the closer to 1, the better

Performance indicator II (target threshold exceedances)

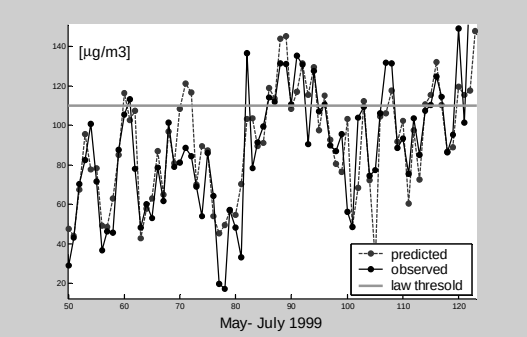
- TPR: True Predicted Rate
- FPR: False Positive Rate
- FA= False Alarm Rate
- SI : success index = TPR – FPR (index of overall ability of the model at predicting whether concentration will exceed or not the threshold)

Predictive performance

	normalized data			deseasonalized data		
	FFNN	**LL**	**PNN**	**FFNN**	**LL**	**PNN**
Average goodness indicators						
ρ	0.83	0.84	0.84	0.83	0.86	0.85
MAE	17.02	15.87	17.13	16.87	15.49	16.41
MBE	-0.70	0.76	-1.03	-0.32	-0.86	-2.13
d	0.90	0.91	0.90	0.90	0.92	0.91
Threshold indicators						
TPR	0.72	0.66	0.66	0.67	0.67	0.73
FPR	0.14	0.10	0.11	0.11	0.10	0.12
FA	0.32	0.28	0.30	0.31	0.30	0.31
SI	0.58	0.56	0.55	0.56	0.57	0.61

- 3 folds cross-validation (1 year each fold).
- Deseasonalization works for LL and PNNs.
- The techniques have similar performance; however LL is better on the average, while PNNs are better around the threshold.

Simulation



Input relevance

- FFNNs: no analysis has been done.
- PNNs: which variables are pruned? The answer is not consistent between the CV runs. However, NO2, atm. Pressure and Pasquill class have been removed in at least 2/3 runs
- LL: since all inputs are standardized, the coefficents are comparable. We have a different parameter set for each query-point. How to analyze them?

LL and input relevance (I)

Avg. parameters estimate grouped according to the solar radiation value:

Solar Rad.	Temp.	Rain	Press.	Stab. class	Humid.	Wind speed
Low solar radiation						
0.221	0.405	-0.529	0.168	-0.075	-0.079	-0.080
Medium solar radiation						
0.304	0.339	-0.230	0.105	-0.030	-0.013	-0.086
High solar radiation						
0.409	0.174	0.043	0.014	0.036	-0.063	0.005

- Press. and Pasquill class: low coefficients (see PNNs)
- Solar radiation, temperature and rainfall as the key variables

LL and input relevance (II)

Avg. parameters estimate grouped according to the solar radiation value:

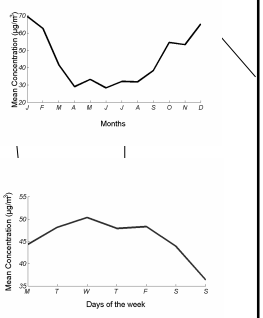
O3 max ($\mu_8(t)$)	O_3 (9a.m.)	CO	NO_2	NO
Low solar radiation				
0.135	0.222	-0.187	0.031	0.461
Medium solar radiation				
0.188	0.242	-0.161	0.064	0.207
High solar radiation				
0.216	0.280	-0.144	0.048	0.042

- CO: negative coefficient (anti-correlated to O3)
- NO2: low coefficient (see PNNs)

Pm10

Pm10 time series analysis

- Available dataset: 1999-2002
- Much higher concentrations in winter (unfavorable dispersion and higher emissions)
- Concentrations are about 25% lower during week-ends
- The application: predict at 9 a.m. the PM10 daily average of today.



Input selection

- We use correlation analysis to define a suitable set of variables and their time aggregators (hourly -> daily)
- The set of inputs is smaller, as PM10 is a primary pollutant and a simpler dynamics compared to O3.

Variable	Aggregation	Variable	Aggregation
PM_{10}	$\mu[22_{t-1} - 8_t]$	*Temp.*	$\mu[4_{t-1} - 8_{t-1}]$
SO_2	$\mu[13_{t-1} - 5_t]$	*Press.*	$\mu[1_{t-1} - 7_t]$

Table 3.6: Input variables chosen for PM_{10} prediction.

- Deseasonalization did not leads to an improvement (but pre-processing via sinusoidal regressor does…)

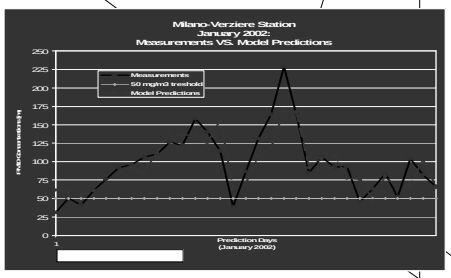
Predictive performance

	FFNN	LL	PNN
Average goodness indicators			
ρ	0.88	0.90	0.89
MAE	8.59	8.25	8.55
MBE	-0.12	0.16	0.47
d	0.94	0.94	0.94
Threshold indicators			
TPR	0.82	0.83	0.83
FPR	0.09	0.08	0.07
FA	0.20	0.17	0.16
SI	0.73	0.75	0.76

- Validation method: cross-validation (1 year each fold).
- Similar performance of the different techniques (moreover, linear regressors are also not very far.)
- Better performance compared to O3 (daily average simpler to predict than 8-hours moving avg, and possibly Pm10 has simpler dynamics than O3)

Model Validation (2002 sample)

- Simulation: January 2002 (with LL)



Conclusions

- **Lazy learning** and PNNs as an interesting and convenient alternative to FFNNs.
- Careful data pre-processing (deseasonalization, input selection) can significantly improve model accuracy.
- Overall, good accuracy achieved on 1-day prediction of both ozone and PM10.
- Lazy learning also allows for a posteriori analysis of the importance of the inputs under different situations.