Machine learning-based uncertainty quantification for data assimilation: a simple model experiment

Maximiliano Sacco[1] (msacco@dc.uba.ar), Yicun Zhen[2], Pierre Tandeo[2], Juan Ruiz[3,4], Manuel Pulido[4,5,6]

[1]University of Buenos Aires, National Meteorological Service of Argentina [2]IMT-Atlantique [3]Atmospheric and Oceanography Research Center (UBA-CONICET) [4]UMI-IFAECI (CNRS-UBA-CONICET) [5]PhysicsDepartment−FACENA−UNNE

Motivation: Quantifying forecast uncertainty is a key aspect of DA systems. In particular we are interested in capturing **state dependent forecast covariance structure**.

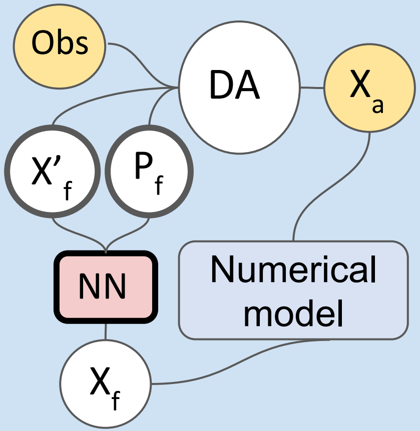
Can neural networks provide a reliable estimation of the forecast error covariance matrix?

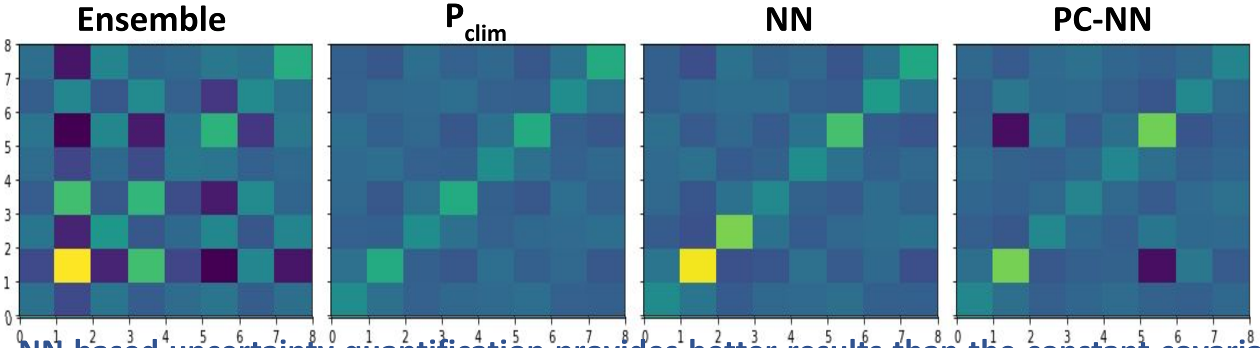
Goal: Design and evaluate machine learning approaches for uncertainty quantification in the context of DA for chaotic dynamical systems.

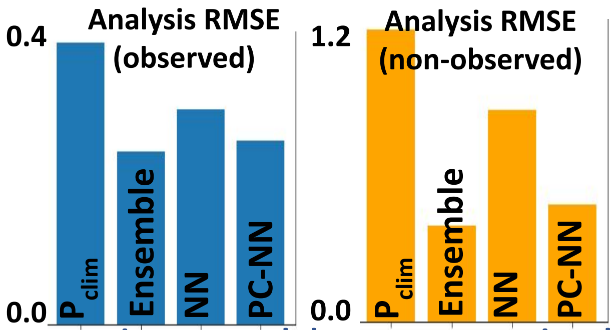
Methods: Build and train a neural network that predicts a state dependent estimation of its forecast uncertainty and corrects the systematic errors in the forecasts.

The NN learns the forecast uncertainty from a dataset of historical forecasts and their corresponding analysis using a likelihood based loss function.

Implementation: Twin experiments with the 2-scale Lorenz model. Fully connected 2-layer network.







NN based uncertainty quantification provides better results than the constant covariance matrix at a much lower computational cost than the ensemble. PC-NN provides better results due to a better representation of covariances (off-diagonal elements). This result in a better estimation of non-observed variables in the PC-NN.

Methodology

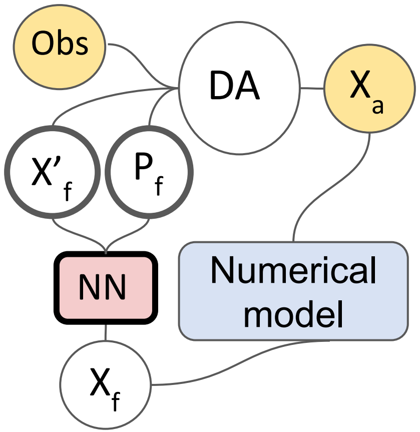
Four DA methods:

Ensemble: A classic EnKF approach using 50 ensemble members. P is estimated as the sample covariance of the forecast ensemble. LETKF is used to compute the analysis update and the ensemble perturbations for the next cycle.

Climatological covariance: A time independent covariance matrix is used combined with a linear Kalman filter type analysis update.

NN: A neural network is used to estimate the variance terms (main diagonal of P_f). Off diagonal terms are estimated assuming a time-independent error correlation matrix. The network also provide a correction for the forecast systematic errors. The input to the network is a deterministic forecast.

NN-PCA: A neural network is used to estimate the covariance matrix in a reduced-dimension space. This is done in an attempt to estimate the most relevant features of the full covariance matrix. The input to the network is a deterministic forecast and the output is an estimation of the full P_f.



Network architecture and training:

Fully connected NN with 2 hidden layers.

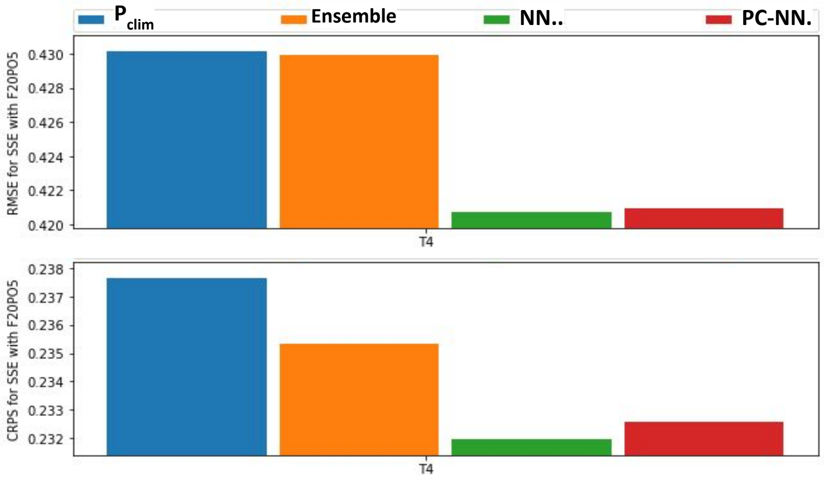
Loss function is based on the likelihood of the targets (analysis) conditioned on the input of the network (short range deterministic forecasts), this allows us to simultaneously estimate a correction for the forecast biases and the forecast uncertainty.

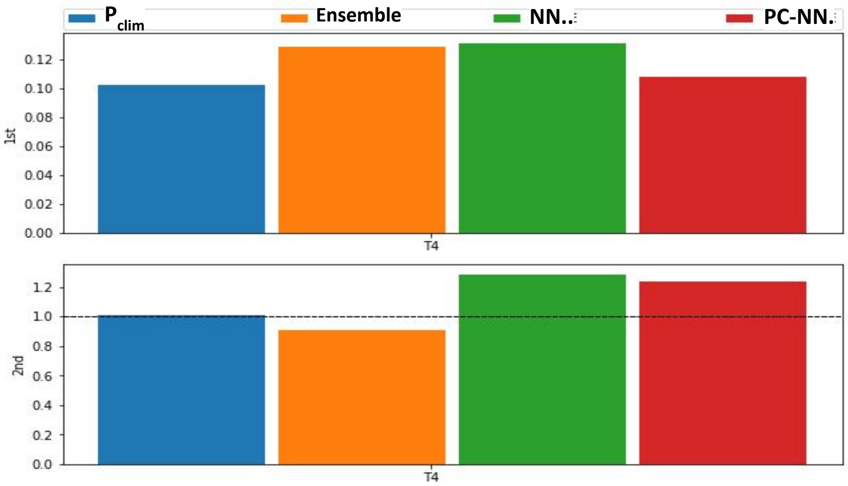
Networks are trained using a large dataset of short-range deterministic forecasts and their corresponding analysis obtained with a 50 member LETKF assimilation cycle. The model used to generate the analysis is an imperfect model (with respect to the model used to generate the observations).

DA Experiments:

Data assimilation experiments are conducted using these 4 methods to provide a first-guess and its associated uncertainty. The system is partially observed (one observation every other grid point). The same imperfect model used to generate the training sample is used in the different data assimilation experiments. Validation is always performed against the nature run.

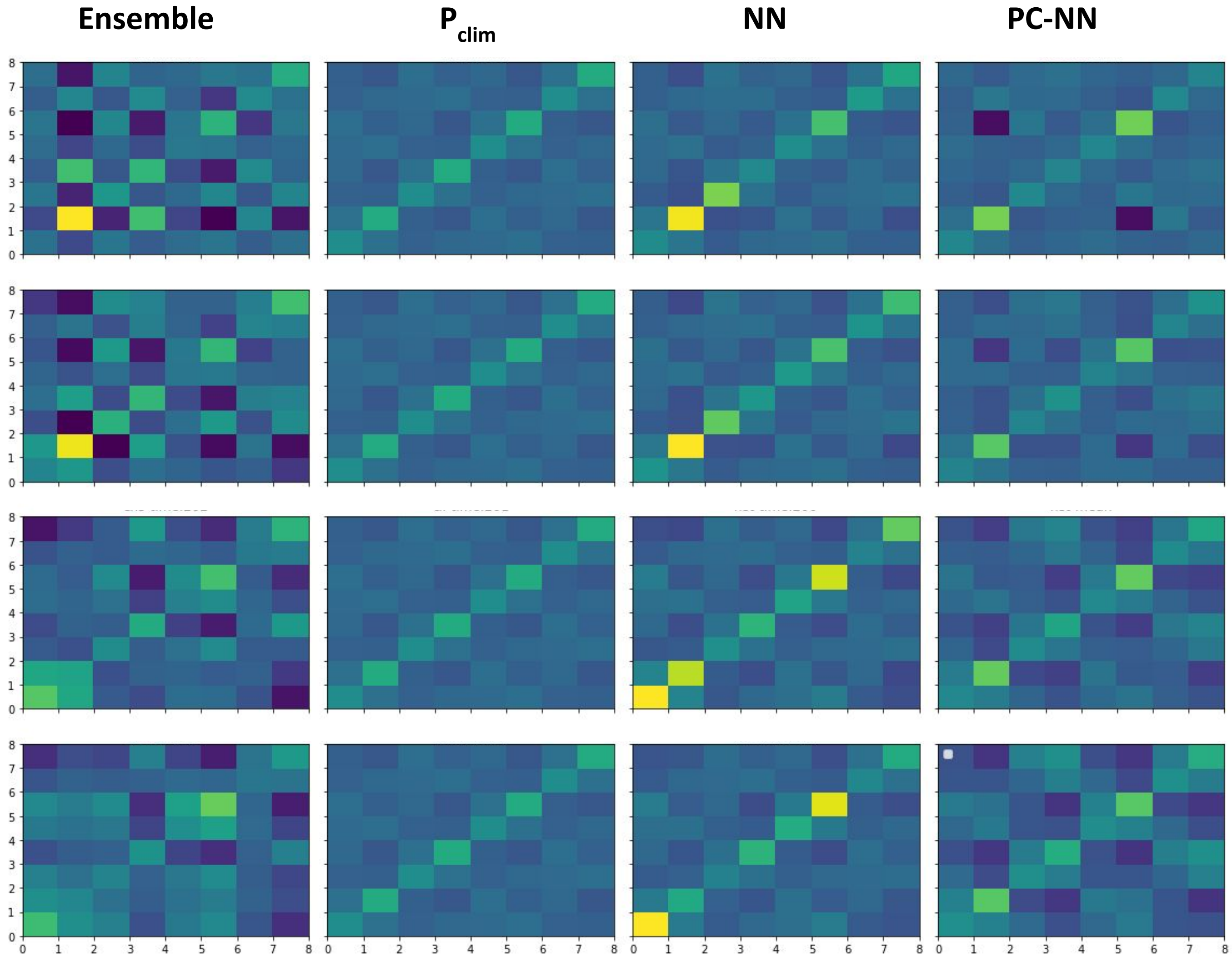
Short-range forecast skill





The figures above show the **RMSE** (root mean square error) on top and **CRPS** (continuous ranked probability score) on bottom for the short-range forecast and its quantified uncertainty. Errors are computed against the Nature Run. Both network perform slightly better than the ensemble in term of accuracy (RMSE) and quantification of the forecast uncertainty (CRPS). The better performance of the network in terms of CRPS is due to the correction of the systematic error performed by the network.

The figures above show the **RCRV** (reduced centered random variable) **1st** and **2nd** moments. The NN has a bias comparable with ensemble and that PC-NN is able to improve the bias the ensemble. The 2nd moment is showing tha both networks are underestimating the uncertainty. This may be a consequence of using analysis as a target in the training of the network. Since forecast and analysis errors are correlated this can lead to an underestimation of the uncertainty.



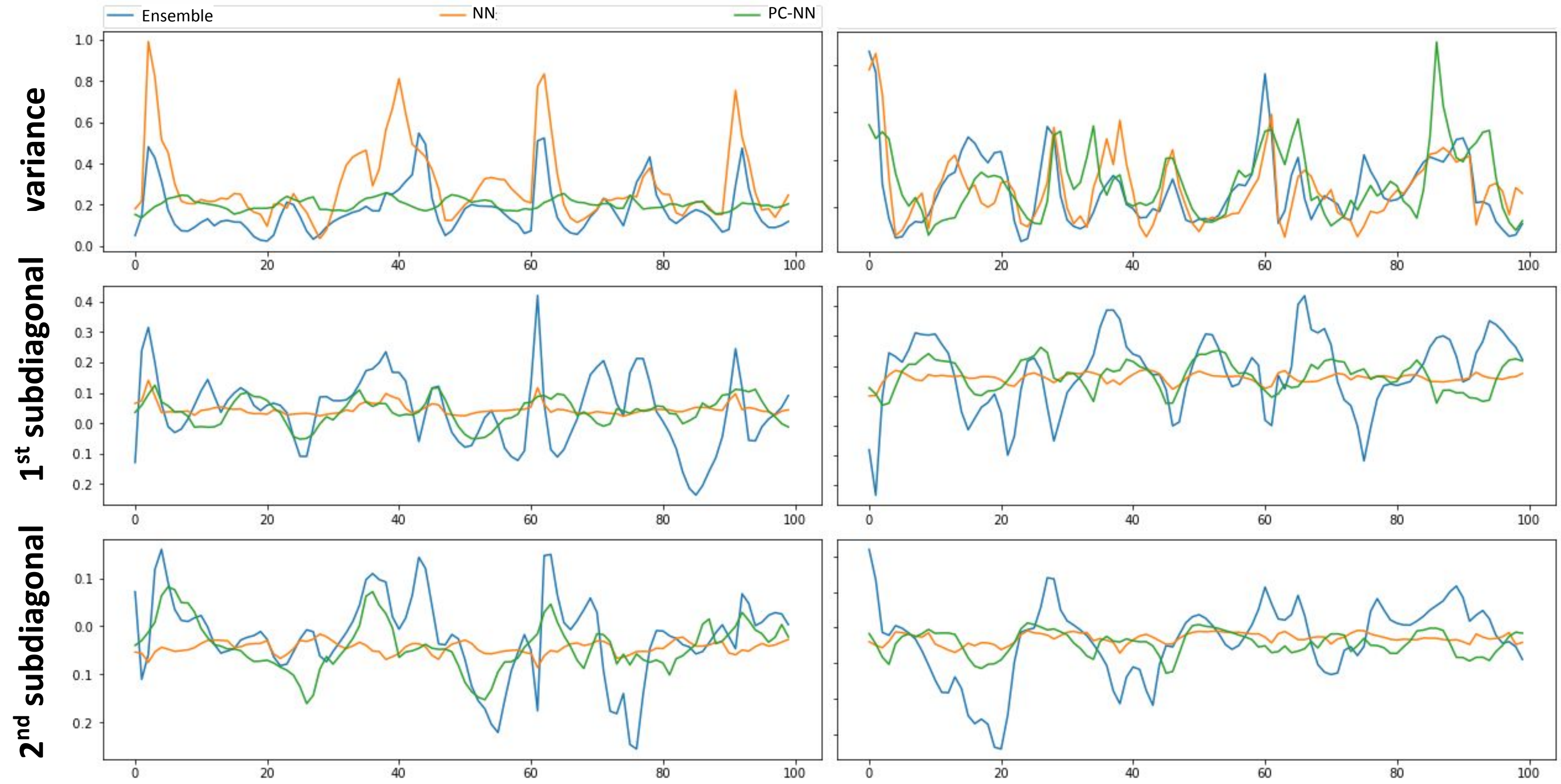
Estimation of covariance Matrix

The figure shows four consecutive times covariance matrix estimated by each methods for a data assimilation cycle with observations every other model grid point.

It is possible to see that both neural networks are able to capture a state-dependant variance (main diagonal) but only PC-NN can adapt to variation on the covariance structure over time.

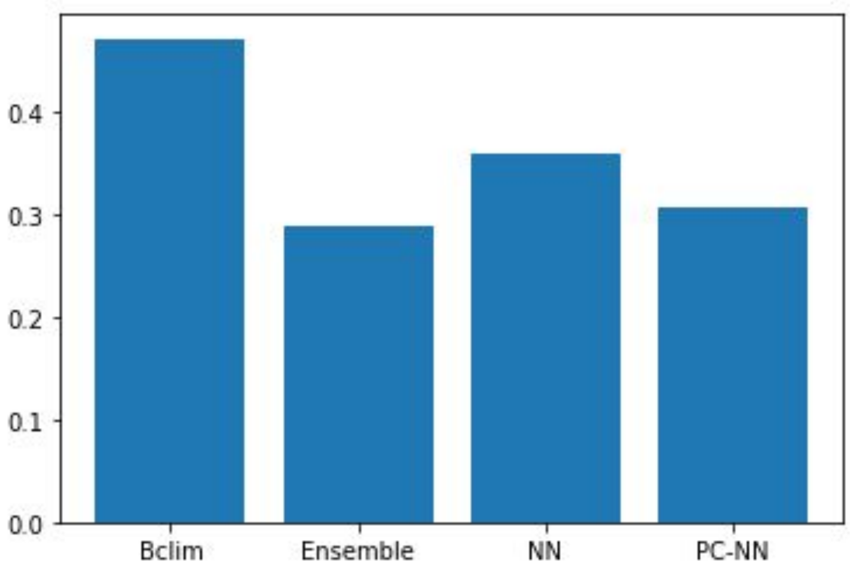
How do the ensemble and the networks estimate the time evolution of forecast error variances and covariances?

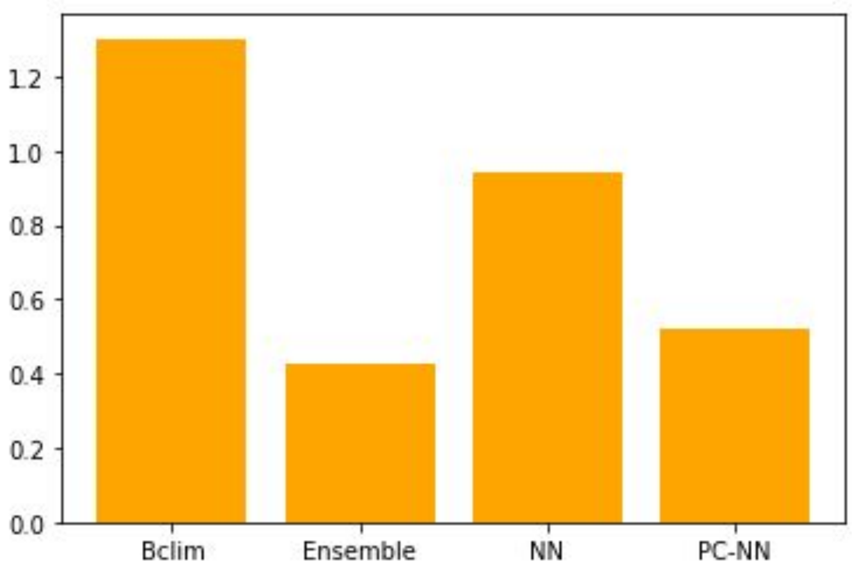
Observed Variables **Non-observed variable**



These plots show the time evolution of variance (top), first (middle) and second (bottom) subdiagonal covariance for an observed (left) and unobserved (right) variable. NN seems to represent better than PC-NN the variance but is unable to reproduce the evolution of covariance as well as PC-NN.

Analysis error





These plots show the **RMSE of analysis** generated with each methodology over 30,000 assimilation cycles. To better visualize the impact of the estimation of forecast error covariances, the RMSE is shown for the observed variables on the left and for the unobserved variables on the right. Both networks are able to improve the P_{clim} baseline method by the estimation of a state-dependant covariance. But only PC-NN has a fully covariance state-dependant information and therefore could improve the error of unobserved variables.