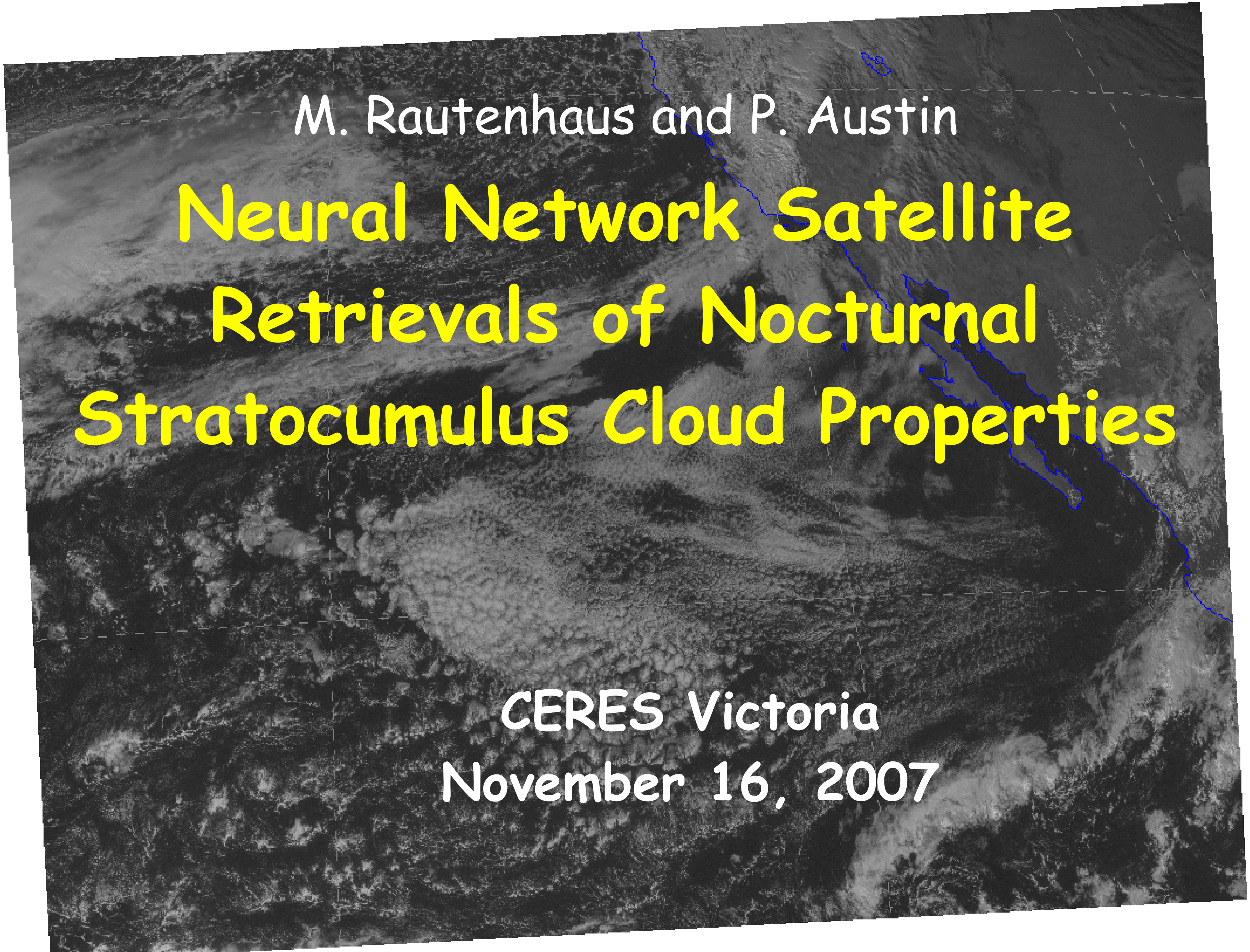


M. Rautenhaus and P. Austin

Neural Network Satellite Retrievals of Nocturnal Stratocumulus Cloud Properties

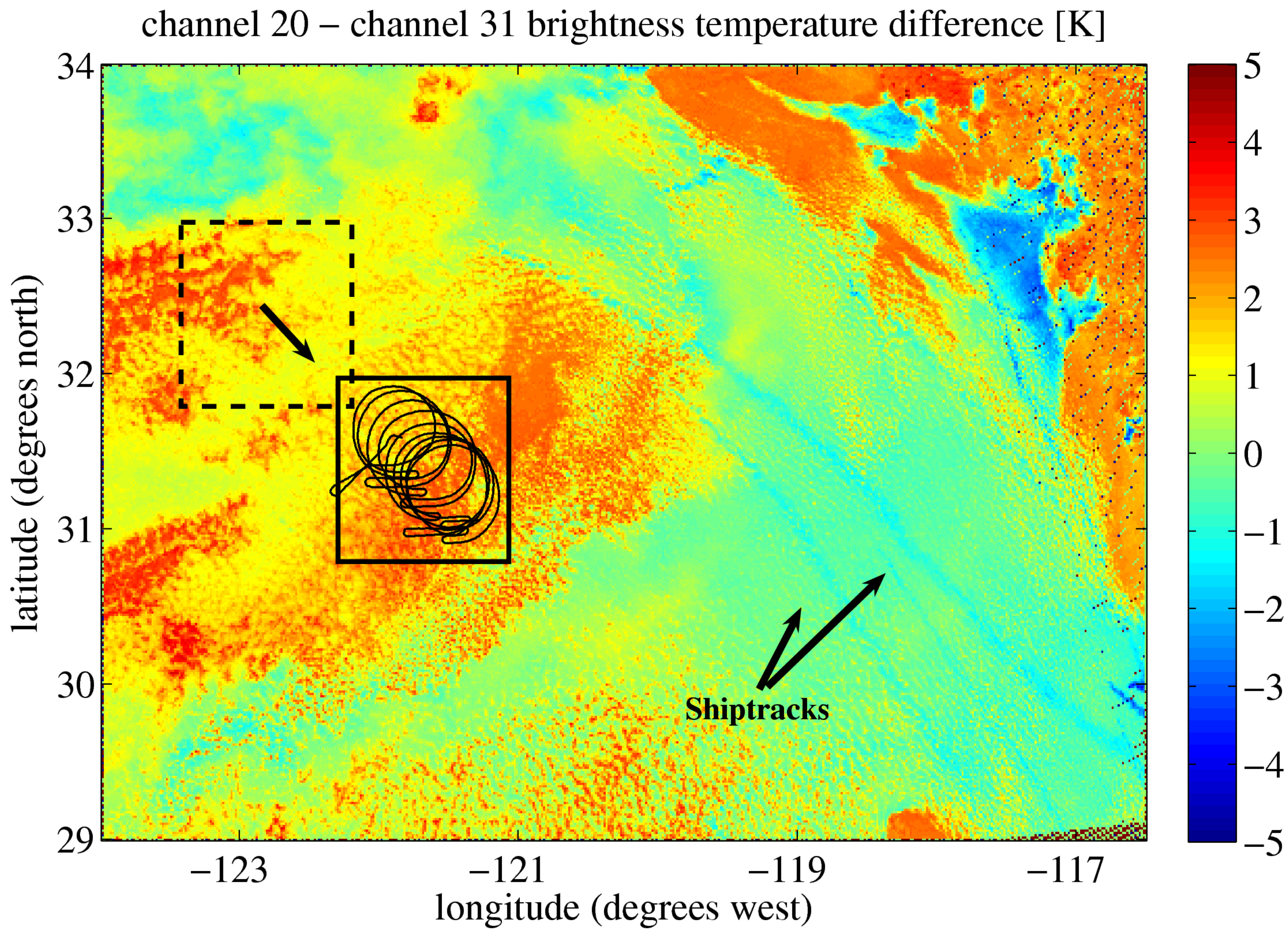
CERES Victoria
November 16, 2007

Outline/Objectives

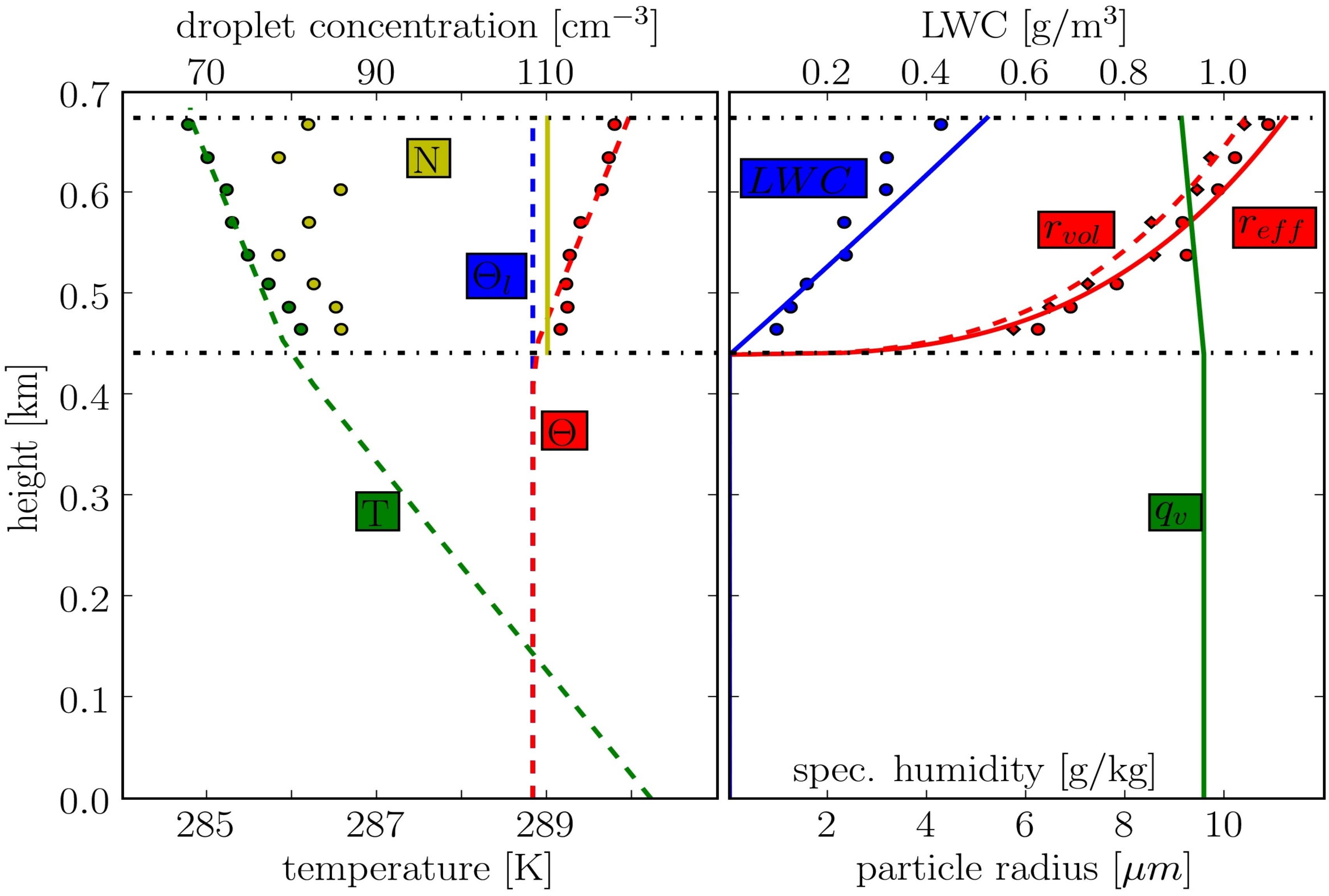
- Determine tau/reff for a nocturnal stratocumulus case with ship tracks and pockets of open cells (DYCOMS II July 11, 2001).
- Use a Bayesian neural network to extract information about uncertainties and sensitivities using the weight distribution of the trained network. (Mackay, 1992, 1995; Aires and Aires et al., 2001, 2002, 2004).

Test Scene: DYCOMS-II RF02, July 11, 2001

- nocturnal flights
- horizontal flight circles at cloud top and bottom
- five hour time lag between satellite overpass and in-situ measurements

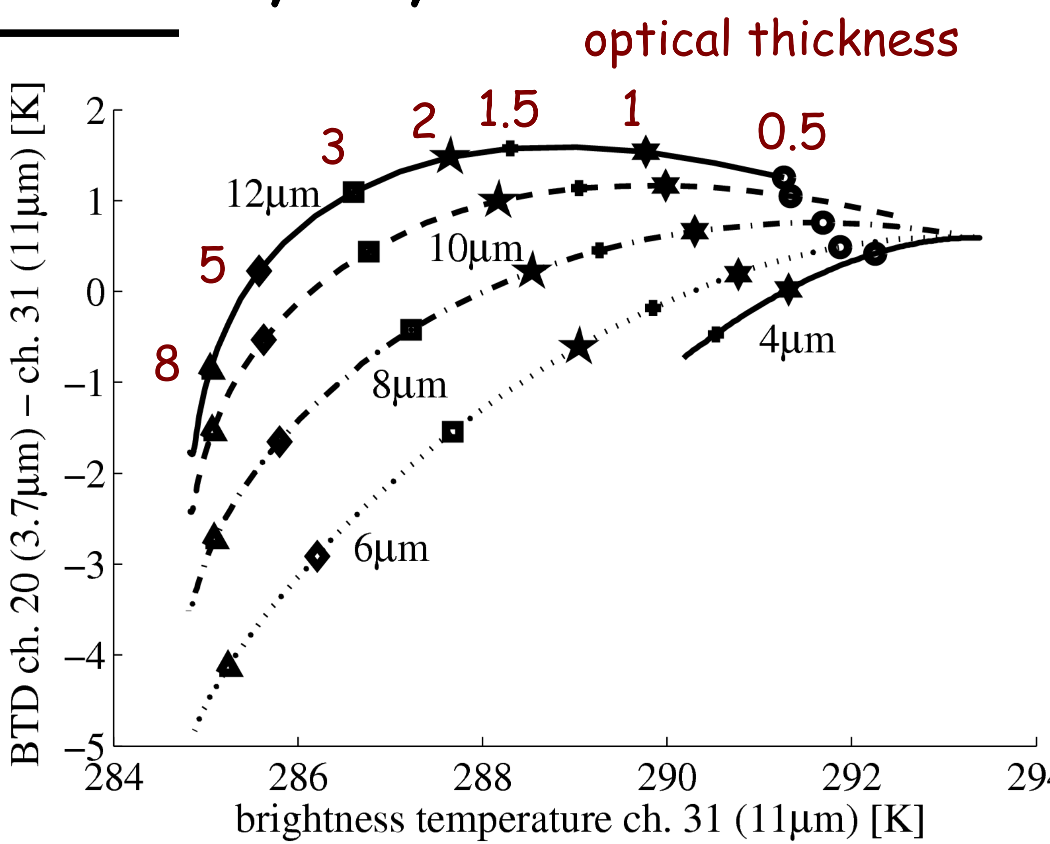


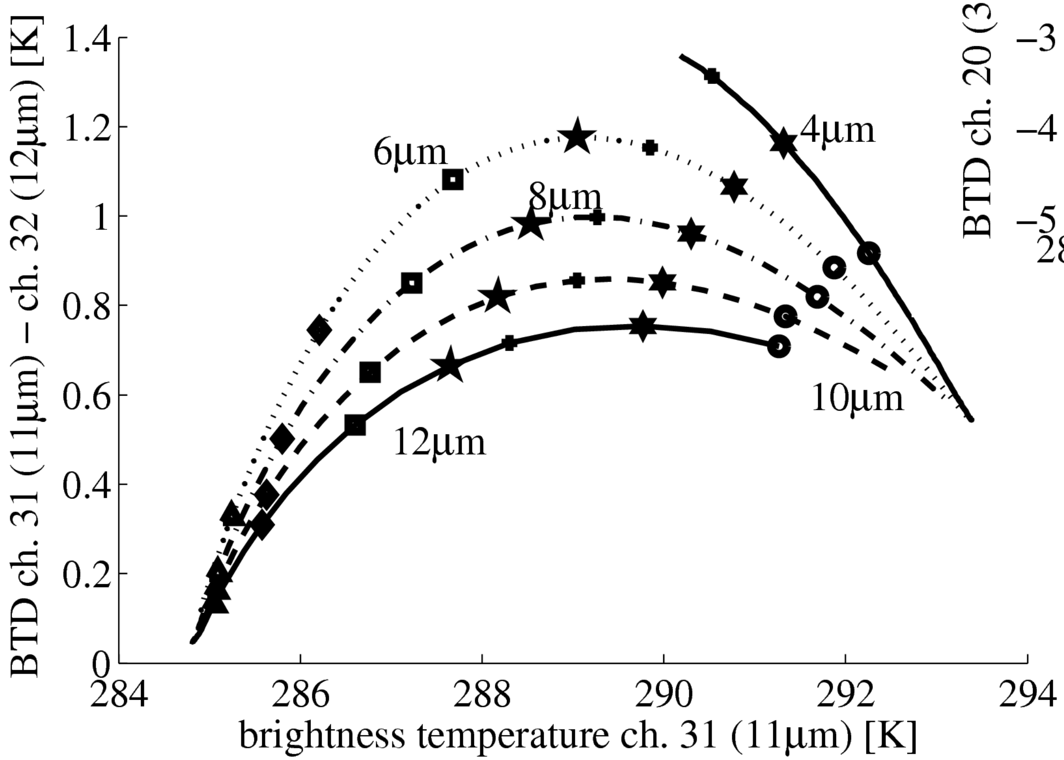
Adiabatic/constant N model compared to DYCOMS sounding



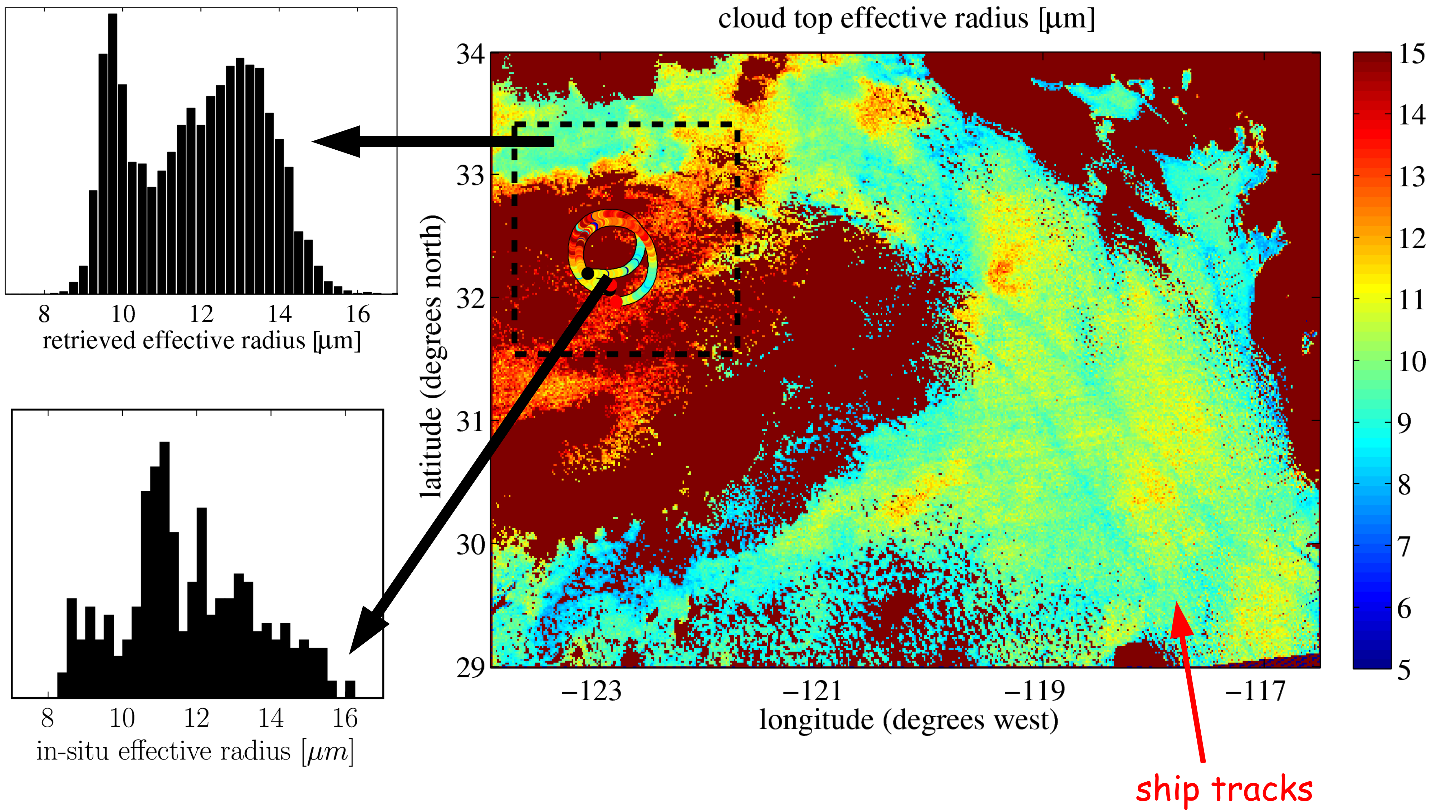
Lookup Table: MODIS Ch. 20, 31, 32

Cloud top temperature = 285 K,
cloud top pressure = 900 hPa.

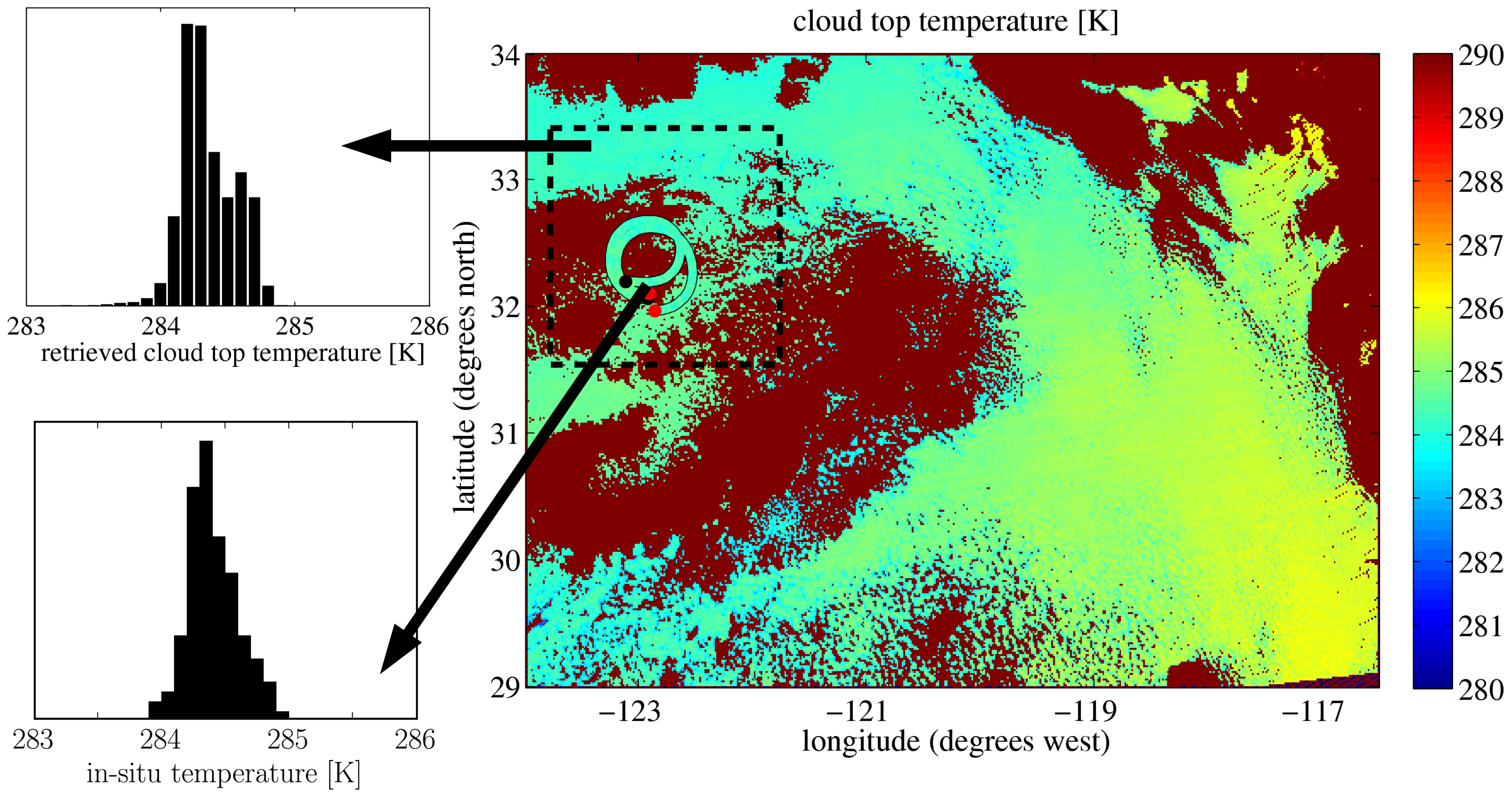




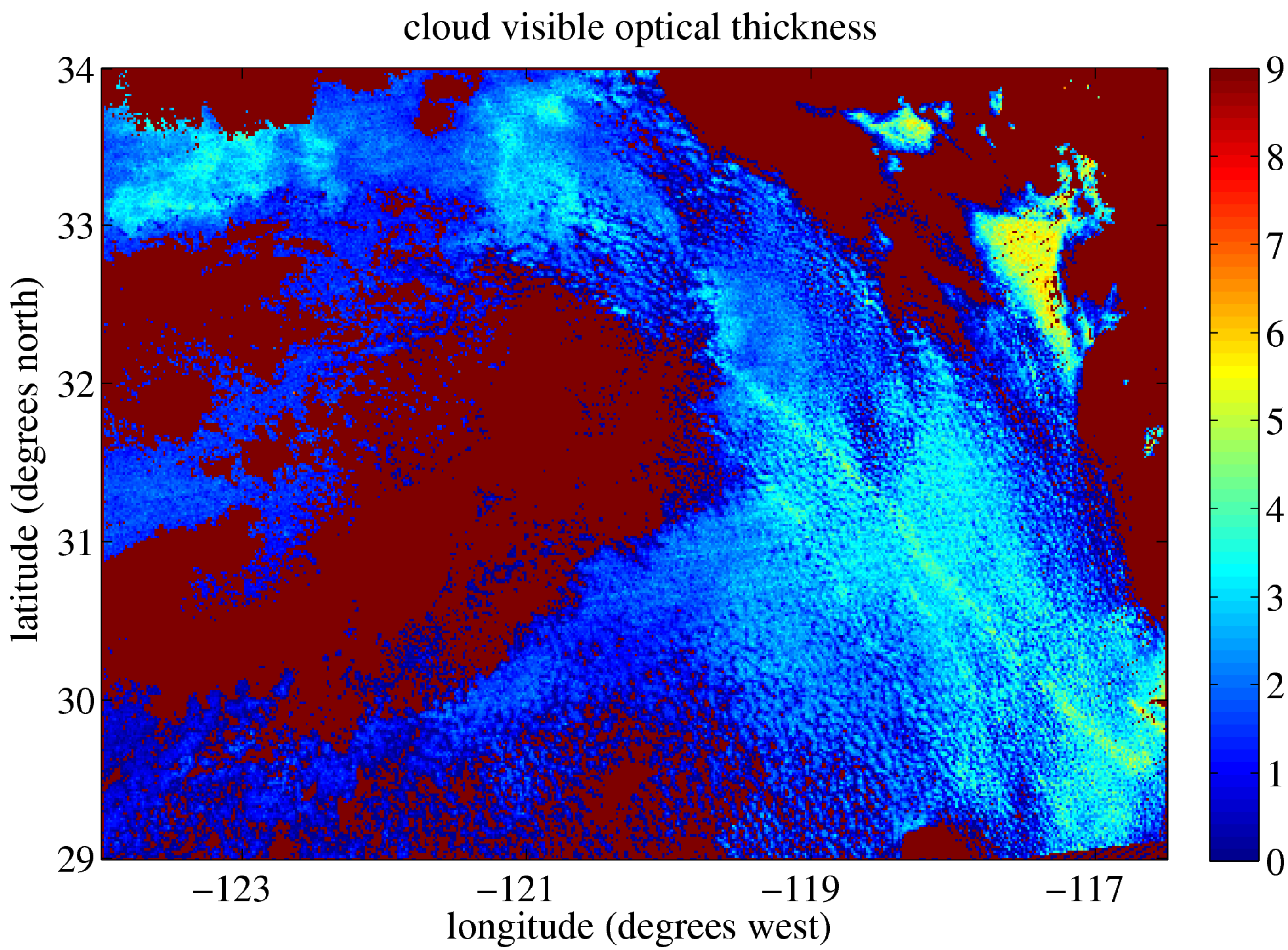
Retrievals: Cloud Top Effective Radius



Retrievals: Cloud Top Temperature



Retrievals: Optical Thickness



Standard retrieval:

- Given a forward radiative transfer model $y(\mathbf{x})$ that maps atmospheric properties

$$\mathbf{x} = \{\tau, r_{eff}, lwp, T_{cld}, \text{overlying atmosphere ...}\} \qquad (1)$$

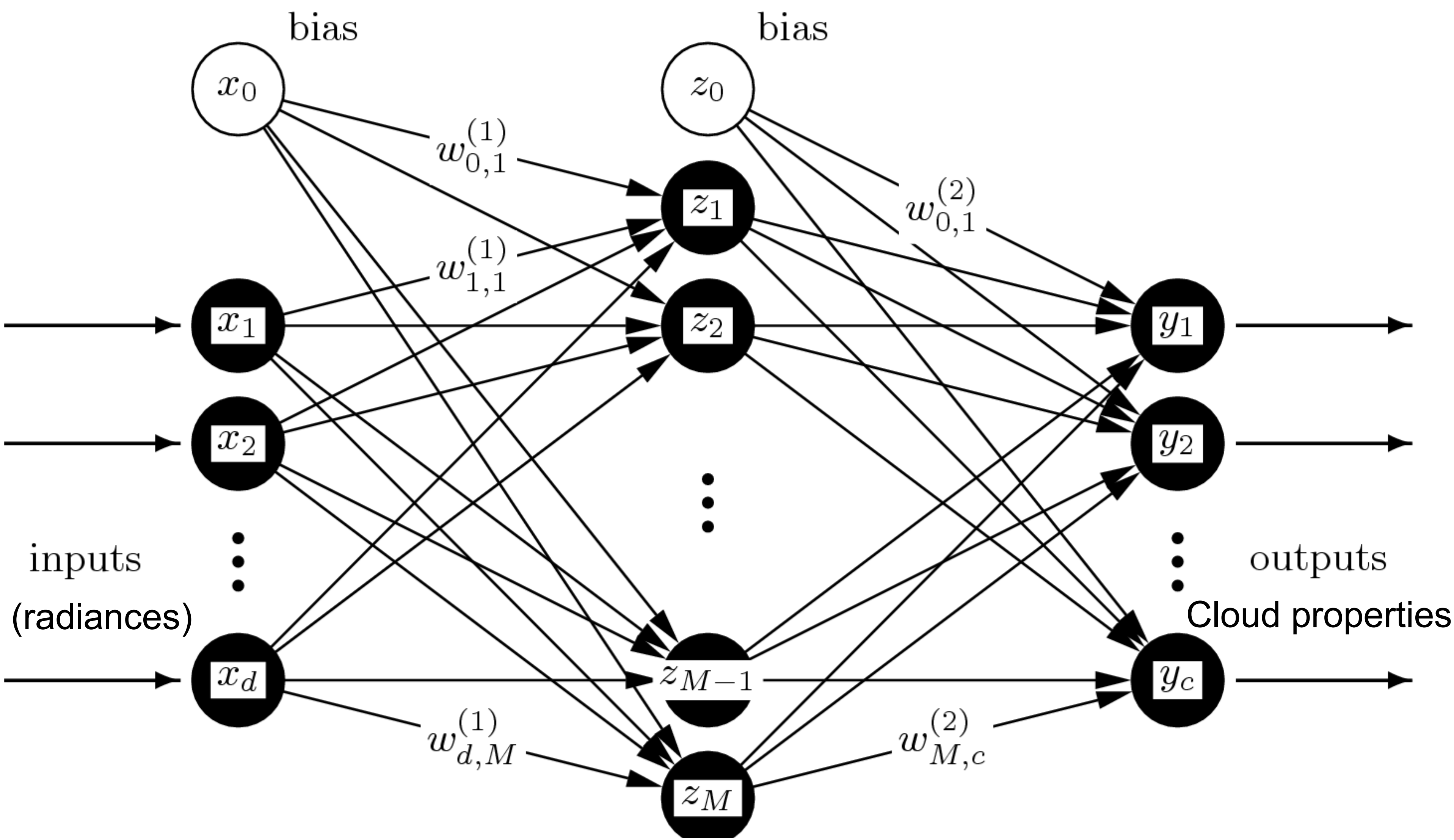
onto radiances (targets) $\mathbf{t}$

$$\mathbf{t} = (I_{3.7},\ I_{11},\ I_{12}, \ldots) = y(\mathbf{x}) \qquad (2)$$

- Find cloud properties $\mathbf{x}_*$ for a radiance measurement $\mathbf{t}_*$ that minimizes a cost function:

$$E(\mathbf{x}) = \sum_{i=1}^{d} (y(x_i) - t_i)^2 \,. \qquad (3)$$

Inversion with Neural Networks



Bayesian neural net:

- Given a training dataset D consisting of TOA radiances $\mathbf{x}$ and cloud targets $\mathbf{t} = \{\tau, r_{eff}, T_{cld}\}$, find the underlying generator for the LUT, y, by choosing a set of network weights w that map $\mathbf{x}$ to $\mathbf{t}$:

$$y_k = \tilde{g}\left(\sum_{j=0}^{M} w_{j,k}^{(2)} \times g\left(\sum_{i=0}^{d} w_{i,j}^{(1)} x_i\right)\right). \tag{4}$$

- The set of optimal weights, w_*, is the one that minimizes the cost function.

$$E = \frac{1}{2}\sum_{n=1}^{N}\{y(x^n; \mathbf{w}) - t^n\}^2. \tag{5}$$

What about Bayes?

- There's no guarantee that the optimal set of weights w_* is the one that gives the best physical representation of the generator y. There is some probability distribution for the weights given the training set, given by Bayes theorem:

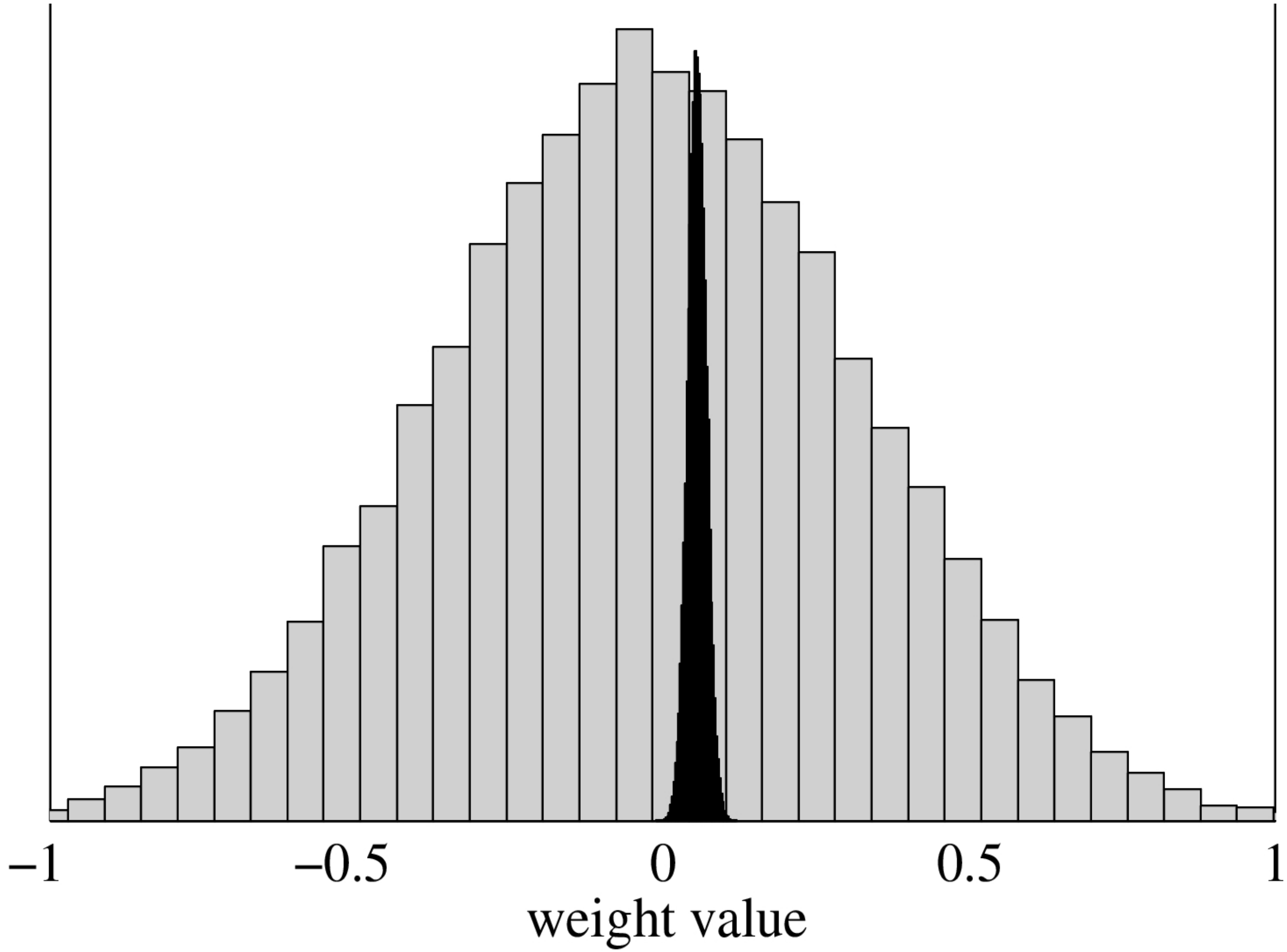
$$p(\mathbf{w}|D) = \frac{p(D|\mathbf{w})\, p(\mathbf{w})}{p(D)}. \tag{6}$$

which can be reduced to the product of two Gaussians:

$$p(\mathbf{w}|D) \propto \exp\left(-\epsilon_D(\mathbf{w}) - \epsilon_W(\mathbf{w})\right), \tag{7}$$

where ϵ_D and ϵ_W are data and weights error functions.

Distribution of W11 for long and short training times



The Hessian and the Jacobian

- Working with $p(\mathbf{w}|D)$: where do we get this PDF?
- Second order Taylor series expansion:

$$\epsilon(\mathbf{w}) = \epsilon(\mathbf{w}^*) + \mathbf{b}^T \cdot \Delta\mathbf{w} + \frac{1}{2}\Delta\mathbf{w}^T \cdot \tilde{\mathbf{H}} \cdot \Delta\mathbf{w},$$

where $\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}^*$. $\mathbf{b}$ denotes the gradient of E at $\mathbf{w}^*$,

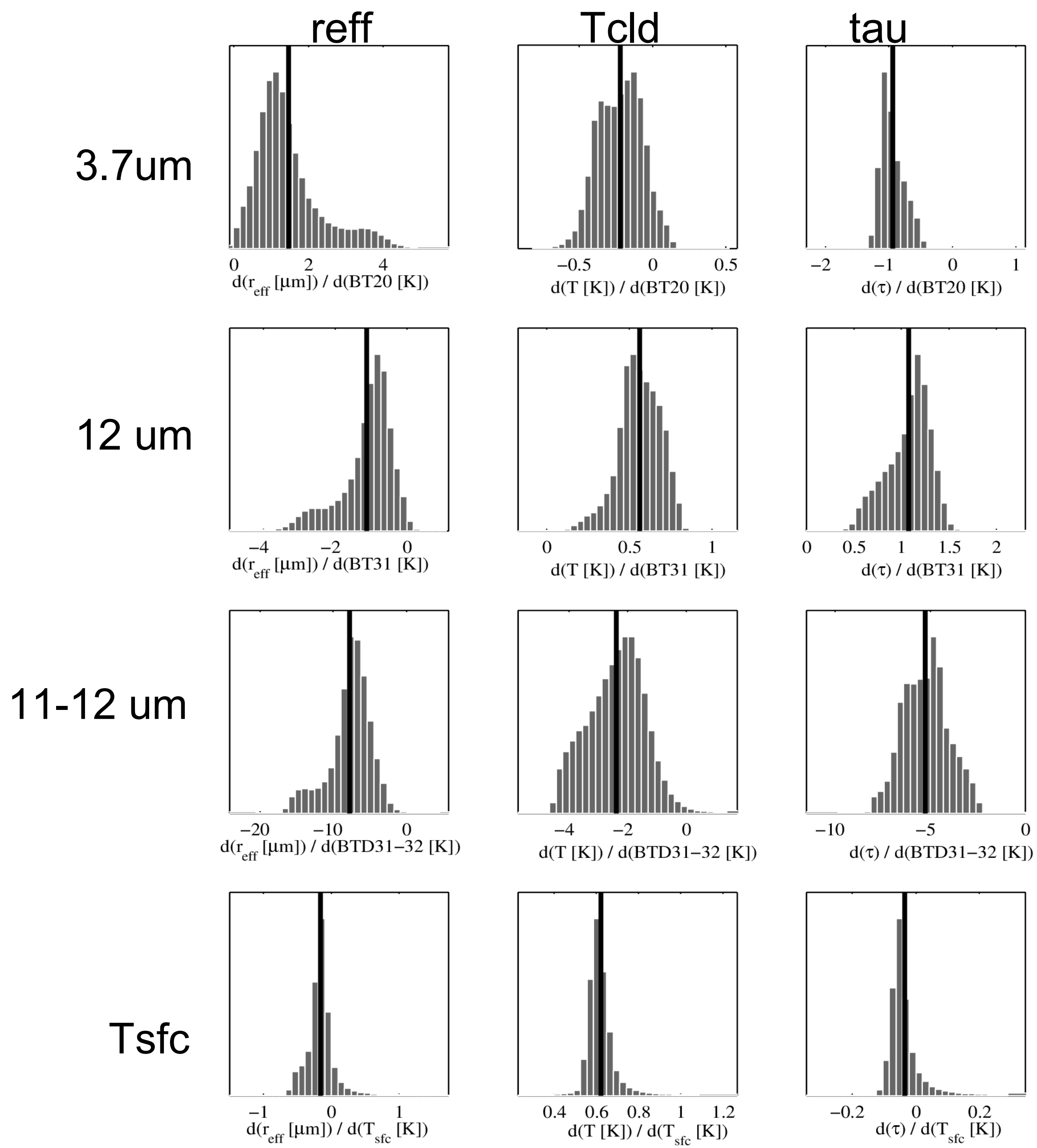
$$\mathbf{b} = \nabla\epsilon(\mathbf{w})|_{\mathbf{w}=\mathbf{w}^*} = \mathbf{0},$$

and the Hessian is given by

$$\begin{aligned} \tilde{\mathbf{H}} &= \nabla\nabla\epsilon(\mathbf{w})|_{\mathbf{w}=\mathbf{w}^*} \\ \text{so that} & \\ p(\mathbf{w}|D) &= \frac{1}{Z}\exp\left(-\frac{1}{2}\Delta\mathbf{w}^T \cdot \tilde{\mathbf{H}} \cdot \Delta\mathbf{w}\right). \end{aligned}$$

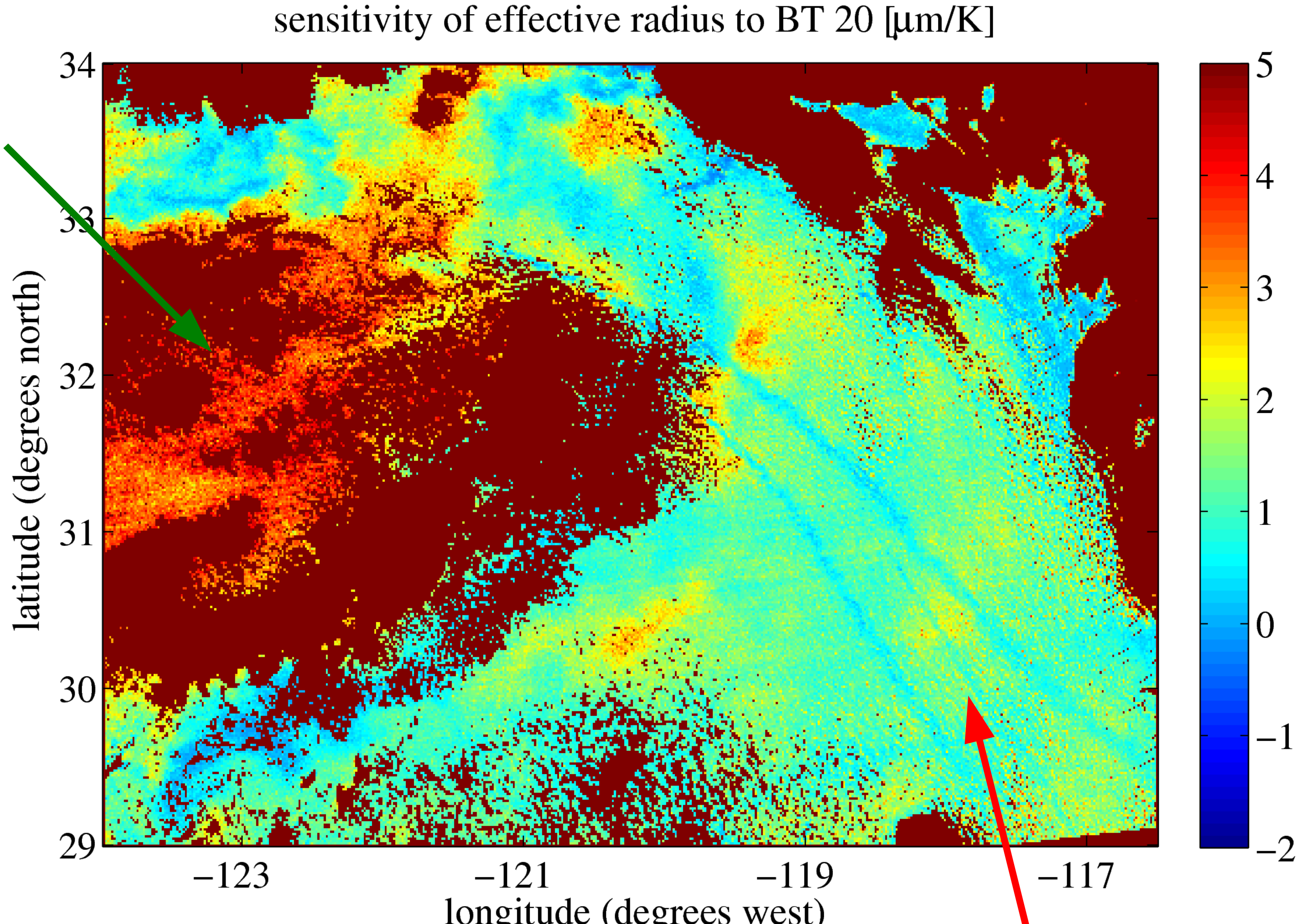
Distribution of the Jacobian for the July 11 scene

$$J_{ij} = \frac{\partial y_j}{\partial x_i}$$



Spatial Distribution of Jacobian $\frac{\partial r_{eff}}{\partial I_{3.7}}$

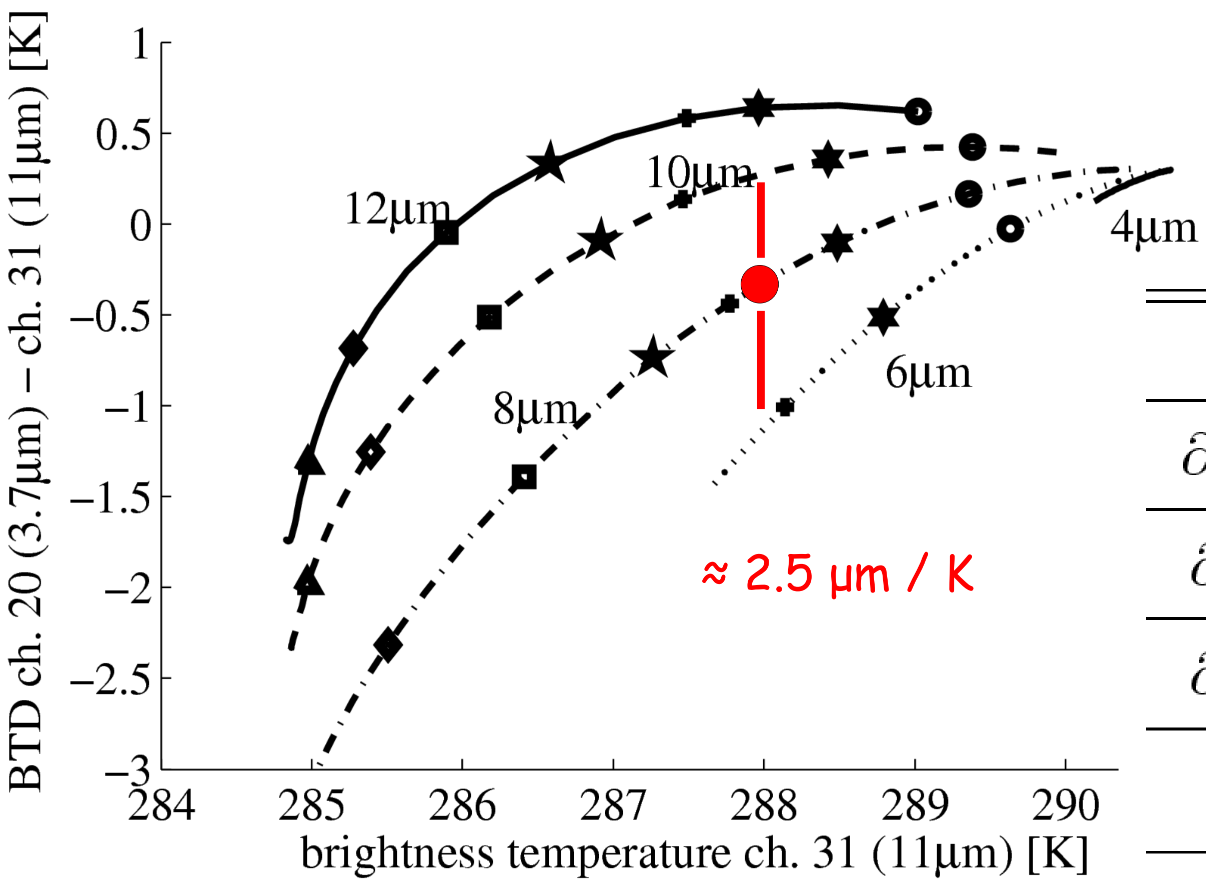
larger droplets, higher sensitivity



smaller droplets, lower sensitivity

Network Architecture

- Jacobian point estimate – compare dependences with "known" values
- Example: brightness temperatures or brightness temperature differences as inputs?



	∂r_{eff} / — [$\mu m/K$]
∂BT(3.7)	0.22 ± 4.72
∂BT(11)	-20.16 ± 20.92
∂BT(12)	19.11 ± 21.67
∂T_{sfc}	1.18 ± 3.37

Scene Jacobian

	$\partial r_{eff}\ /\ $— $[\mu m/K]$	$\partial T\ /\ $— $[K/K]$	$\partial \tau\ /\ $— $[K^{-1}]$
∂BT(3.7)	1.47 ± 1.32	-0.22 ± 0.56	-0.94 ± 1.53
∂BT(11)	-1.23 ± 1.37	0.56 ± 0.80	1.07 ± 1.69
∂BTD(11-12)	-7.75 ± 6.11	-2.39 ± 2.32	-5.24 ± 7.86
∂T_{sfc}	-0.16 ± 0.54	0.62 ± 0.43	-0.04 ± 0.82

- Use mean Jacobian to estimate

    1) average dependences
    2) ill-conditioning of the problem
    3) sensitivities to inputs
    4) importance of inputs

normalised mean Jacobian – importance of inputs

	∂r_{eff} / — [μm]	∂T / — [K]	$\partial \tau$ / — [1]
∂BT(3.7)	2.36	−0.35	−1.51
∂BT(11)	−1.82	0.91	1.74
∂BTD(11-12)	−1.34	−0.42	−0.92
∂T_{sfc}	−0.20	0.78	−0.04

Summary

- Neural net able to retrieve reff, Tcld, tau for nocturnal stratocumulus case
- Bayesian approach uses the Hessian of the error function to estimate weight distribution, distribution of the network sensitivities (Jacobian)
- More work to do on generating a robust network (network ensembles), improving the retrieval via the prior weight distribution, tracking diurnal changes.