Learning the interpolator for snow depth and snow water equivalent

Let's describe s_d – measured snow depth, s_w – snow water equivalent from SNOW-E model.

The interpolator from set of points $\vec{x}_j$ to set of points $\vec{y}_j$ can be written as:

$$s_X\left(\vec{y}_j\right) = \sum_{i \in I_j} w_X(i,j) s_X\left(\vec{x}_i\right),$$

where I_j – set of influence points, $w_X(i,j) \geq 0$ – optimized weights, $X = d, w$.

For search optimal w_X we use the principle: the minimal MSE for interpolation to excluded station:

$$L(s_X)^2 = \frac{1}{N} \sum_{j=1}^{N} \left(s_X\left(\vec{x}_j\right) - \sum_{i \in I_j \setminus \{j\}} w_X(i,j) s_X\left(\vec{x}_i\right) \right)^2 \to \min_{w_X}$$

The interpolator takes into account the altitudinal zonality

We suggest then the snow quantity depends on the altitude over sea level *h:*

$$s_X\left(\vec{x}_j\right)=s_X\left(\vec{x}_i\right)\exp\left(c_X\left(h\left(\vec{x}_j\right)-h\left(\vec{x}_i\right)\right)\right)+\varepsilon_{ijX},$$

where c_X – skew-symmetric function, ε_{ijX} – random noise.

Then we can consider the altitudinal depended interpolator:

$$s_X\left(\vec{y}_j\right)=\frac{\sum_{i\in I_j}\exp\left(\omega_{ij}+c_X\left(h\left(\vec{y}_j\right)-h\left(\vec{x}_i\right)\right)\right)s_X\left(\vec{x}_i\right)}{\sum_{i\in I_j}\exp\left(\omega_X\left(i,j\right)\right)}$$

The simplest case: the exponential smoothing altitudinal depended interpolator:

$\omega_{ij}=-\lambda d\left(\vec{y}_j,\vec{x}_i\right),\ c_X\left(\Delta h\right)=\alpha\Delta h,$ where $d\left(\vec{y}_j,\vec{x}_i\right)$ – distance, $\alpha,\lambda>0$ – optimized constants.

Optimization results: interpolate to excluded station

Model	Parameters	Depended subsample 01/09/2019-16/08/2020, cases: 228284		Undepended subsample 01/09/2020-14/11/2020, cases: 20618	
RMSE interpolation error*		$L(s_w)$, **kg/m²**	$L(s_d)$, **cm**	$L(s_w)$, **kg/m²**	$L(s_d)$, **cm**
$\omega_X = -\lambda d,\ c_X \equiv 0$	1	47.5	17.6	20.7	9.5
$\omega_X = -\lambda d,\ c_X = \alpha_X \Delta h$	3	47.4	17.6	20.6	9.5
$\omega_X = NN,\ c_X \equiv 0$	192	43.5	16.4	16.3	8.1
$\omega_X = NNN,\ c_X \equiv 0$	256	42.2	15.6	15.9	7.9
$\omega_X = NN,\ c_X = sNN$	387	42.9	16.3	16.2	7.9
$\omega_X = NNN,\ c_X = sNN$	451	41.4	15.4	15.0	7.4

Where: NN – neural network, depends on $d, \Delta h$. NNN – neural network, depends on $d, \Delta h,\ s_w(\vec{x}_i), s_h(\vec{x}_i)$,

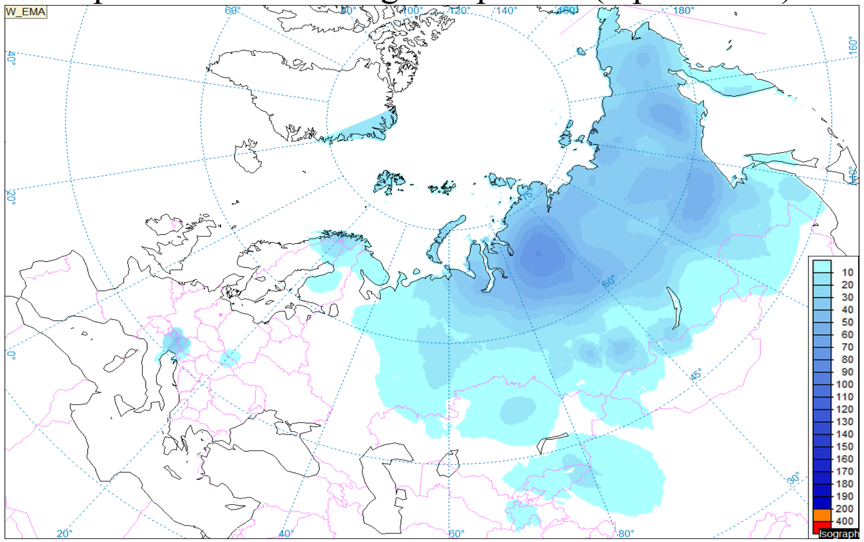
$sNN = f_{NN}(\Delta h) - f_{NN}(-\Delta h), f_{NN}$ – neural network.

**We compare only nonzero amounts of snow: need to refine the snow mask using remote sensing of the Earth data.*

Example: 2020/11/14 snow analysis

Exponential smoothing interpolator (1 parameter)

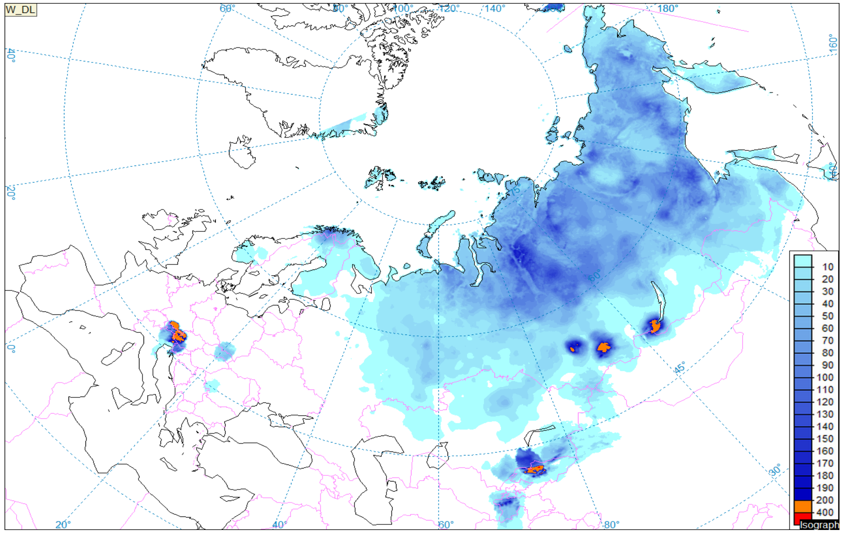
Deep learned interpolator (neural network with 451 parameter)

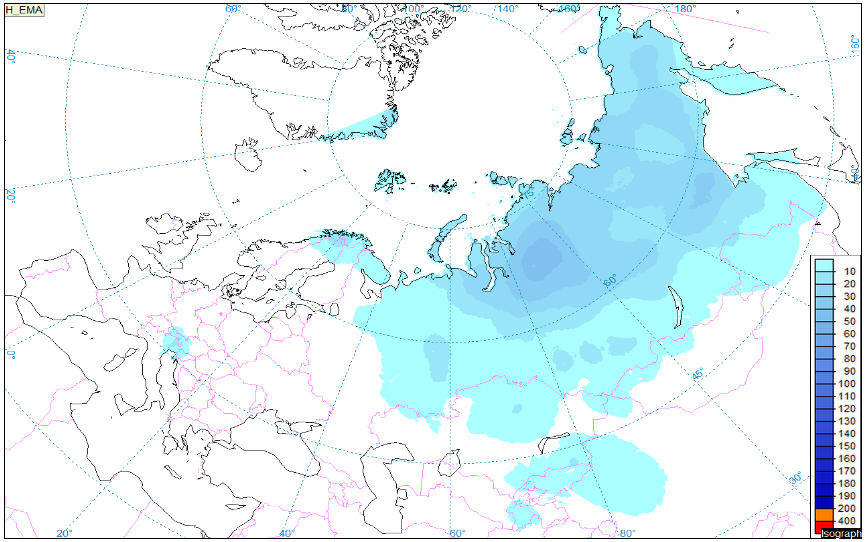


snow

water

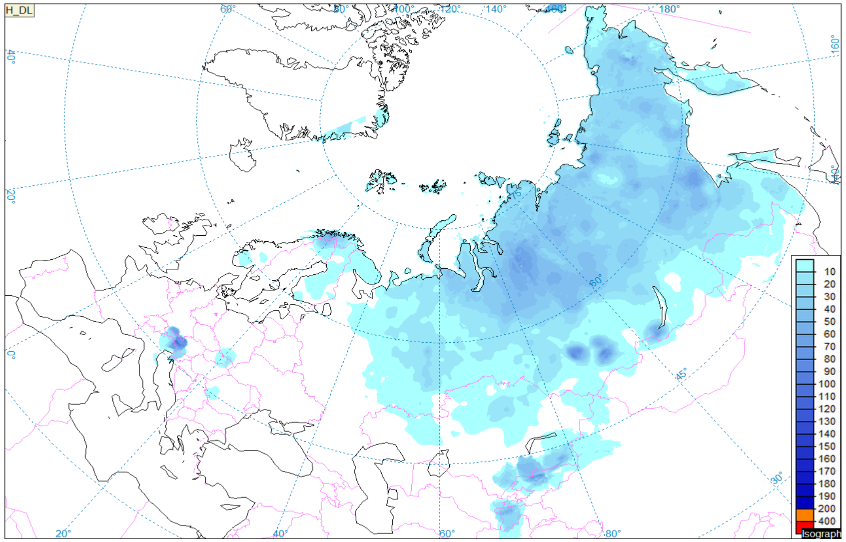
content





snow

depth



Conclusions

1. We offer the deep learning approach for interpolation the quantity of snow problem
2. We take into account the altitudinal zonality
3. The deep learned interpolator is more accurate and higher detailed then exponential smoothing
4. Our approach is easy extended: can take into account other geographical (quantity of forests, etc.) and other (time of year, first guess, etc.) parameters
5. The method needs to refine the snow mask using remote sensing of the Earth data

Thank you for attention!