AE IMS135, Assignment on Machine Learning for Turbulence Modeling

Machine Learning will be used to improve the non-linear $k-\varepsilon$ model of Craft *et al.*. The model for the Reynolds stresses read (see Eq. 14.3)

$$
\begin{aligned}
a_{ij} &\equiv \frac{\overline{v_i'v_j'}}{k}-\frac{2}{3}\delta_{ij}=\boxed{-2c_\mu\tau\bar{s}_{ij}} \\
&+ c_1\tau^2\left(\bar{s}_{ik}\bar{s}_{kj}-\frac{1}{3}\bar{s}_{\ell k}\bar{s}_{\ell k}\delta_{ij}\right)+c_2\tau^2\left(\bar{\Omega}_{ik}\bar{s}_{kj}-\bar{s}_{ik}\bar{\Omega}_{kj}\right) \\
&+ c_3\tau^2\left(\bar{\Omega}_{ik}\bar{\Omega}_{jk}-\frac{1}{3}\bar{\Omega}_{\ell k}\bar{\Omega}_{\ell k}\delta_{ij}\right)+c_4\tau^3\left(\bar{s}_{ik}\bar{s}_{k\ell}\bar{\Omega}_{\ell j}-\bar{\Omega}_{i\ell}\bar{s}_{\ell k}\bar{s}_{kj}\right) \\
&+ c_5\tau^3\left(\bar{\Omega}_{i\ell}\bar{\Omega}_{\ell m}\bar{s}_{mj}+\bar{s}_{i\ell}\bar{\Omega}_{\ell m}\bar{\Omega}_{mj}-\frac{2}{3}\bar{\Omega}_{mn}\bar{\Omega}_{n\ell}\bar{s}_{\ell m}\delta_{ij}\right) \\
&+ c_6\tau^3\bar{s}_{k\ell}\bar{s}_{k\ell}\bar{s}_{ij}+c_7\tau^3\bar{\Omega}_{k\ell}\bar{\Omega}_{k\ell}\bar{s}_{ij}
\end{aligned}
\tag{AE.1}
$$

$$
\bar{s}_{ij}=\frac{1}{2}\left(\frac{\partial\bar{v}_i}{\partial x_j}+\frac{\partial\bar{v}_j}{\partial x_i}\right),\quad \bar{\Omega}_{ij}=\frac{1}{2}\left(\frac{\partial\bar{v}_i}{\partial x_j}-\frac{\partial\bar{v}_j}{\partial x_i}\right),\quad \tau=k/\varepsilon \tag{AE.2}
$$

The constants are set to [59] (see also p. 177)

$$
\begin{aligned}
c_1 &= -0.05,\quad c_2=0.11,\quad c_3=0.21,\quad c_4=-0.8 \\
c_5 &= 0,\quad c_6=-0.5,\quad c_7=0.5
\end{aligned}
\tag{AE.3}
$$

In fully developed channel flow ($\bar{v}_2=\bar{v}_3=\partial/\partial x_1=\partial/\partial x_3\equiv 0$), Eq. AE.1 is simplified as [220]) (recall that $-c_5+c_6+c_7=0$)

$$
\overline{v_1'^2} = \frac{k}{12}\tau^2\left(\frac{\partial\bar{v}_1}{\partial x_2}\right)^2(c_0+6c_2)+\frac{2}{3}k \tag{AE.4}
$$

$$
\overline{v_2'^2} = \frac{k}{12}\tau^2\left(\frac{\partial\bar{v}_1}{\partial x_2}\right)^2(c_0-6c_2)+\frac{2}{3}k \tag{AE.5}
$$

$$
\overline{v_3'^2} = -\frac{k}{6}\tau^2\left(\frac{\partial\bar{v}_1}{\partial x_2}\right)^2c_0+\frac{2}{3}k \tag{AE.6}
$$

$$
\overline{v_1'v_2'} = -c_\mu\tau\frac{\partial\bar{v}_1}{\partial x_2}
$$

where I denote $c_0\equiv c_1+c_3$.

Now we want to compute the coefficients in this equation system using Machine Learning. In order to create a target (i.e. the exact coefficient), we first compute the coefficients using DNS data. Hence we invert the equation system and we get [220]

$$
c_0=-\frac{6a_{33}}{\tau^2\left(\frac{\partial\bar{v}_1}{\partial x_2}\right)^2},\quad c_2=\frac{2a_{11}+a_{33}}{\tau^2\left(\frac{\partial\bar{v}_1}{\partial x_2}\right)^2} \tag{AE.7}
$$

AE.1 Machine Learning

The objective is now to use Machine Learning to adapt the coefficients c_0 and c_2 so that the normal stresses using Eqs. AE.4, AE.5 and AE.6 agree with DNS of developing boundary layer at $Re_\theta = 8\,180$ [41]. The mean velocity, $\bar{v}$, k and ε in Eqs. AE.4 – AE.6 are taken from DNS [41]. The DNS data non-dimensional with u_τ and ν.

Support Vector Machines (SVR) allows only one output parameter. Here we have two output parameters, c_0 and c_2. Hence, we choose to use Neural Network and we will use `pytorch` in Python.

The next question is which input variable should be used. Looking at Eq. AE.4 – AE.6, the square of the velocity gradient seems to be an obvious choice. Note that the input variables should be non-dimensional so that the Neural Network Model can be used to predict other flow cases and Reynolds numbers. As mentioned above, the DNS data are already non-dimensional. Recall that the velocity gradients must be taken with respect to y^+ (not y) in order to get a non-dimensional velocity gradient. We probably need at least one more input variable. There are a number of different suitable choices. Here are some of the combinations of input variables I tried

$$\left(\partial U^+/\partial y^+\right)^2 \quad , \quad \left(\partial U^+/\partial y^+\right)^{-1} \tag{AE.8}$$

$$T^2\left(\partial U/\partial y\right)^2 \quad , \quad T\left(\partial U/\partial y\right)^{-1}, \quad T = k/\varepsilon \tag{AE.9}$$

$$\left(\partial U^+/\partial y^+\right)^2 \quad , \quad k^+/\varepsilon^+ \tag{AE.10}$$

Additional input variables maybe

$$\nu_t^+ \equiv \nu_t/\nu \tag{AE.11}$$

$$\frac{\nu_t}{L^2 T}, \quad L = \frac{k^{3/2}}{\varepsilon}, \quad T = \frac{k}{\varepsilon} \tag{AE.12}$$

$$y^+ \tag{AE.13}$$

The advantage of Eq. AE.9 is that the scaling is local; the other options depend on scaling based on wall quantities which may be difficult to use in complex geometries. Here I use Eq. AE.9 (Eq. AE.8 gives virtually identical results).

Here you find one Python script

- `NN-train-BL.py`. This script loads the DNS data of boundary-layer flow [41, 221]

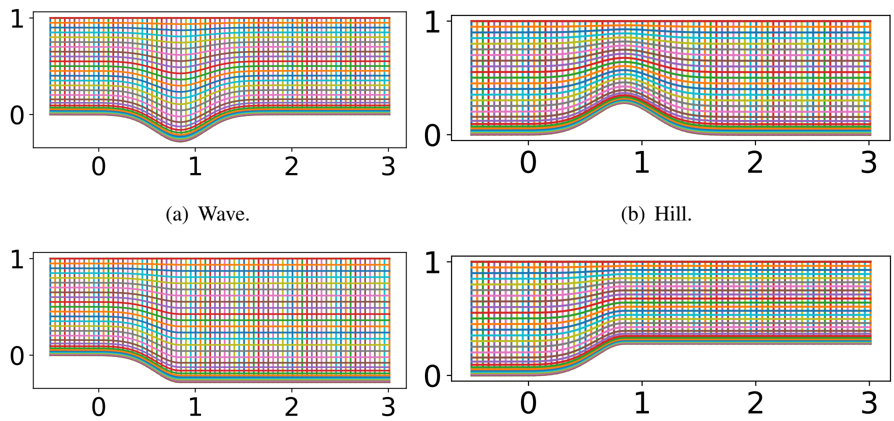
On the www page you also find the DNS data files.

AE.2 The Assignment

Assignment 1 In Section AE.1 the NN model is trained on the normal Reynolds stresses in the boundary layer flow and then the Reynolds stresses in the two channel flows ($Re_\tau = 550$ and $Re_\tau = 5\,200$) are predicted. Do it the other way around: use one of the channel flows as training data and predict the Reynolds stresses for the other two flows. You need probably change the limit on y^+ and $\partial U^+/\partial y^+$.

Assignment 2a Use different different input parameters when training the NN in one of the boundary layer flow/channel flows. Try to get better agreement in the other two flows that I got [220].

Assignment 2b Instead of using Neural Network, NN (Python's `pytorch`) use Random Forest, RF (Python's `RandomForestRegressor`)



(a) Wave. (b) Hill.

(c) Expansion. (d) Contraction.

Figure AE.1: Grids for DNS simulations. Every 5^{ths} grid lines are shown.

Assignment 2c Evaluate the NN model in one or many of the flow in Fig. AE.1. Then you need to make some assumptions for the coefficients c_4, c_6 and c_6: should they stay constant? should their ratio to c_0 and/or c_2 stay constant? or ...

Assignment 2d Do something else!

Next step? (MSc thesis project) Carry out a CFD simulation with e.g. STAR-CCM+, ANSYS or OpenFOAM using the NN predict script. Then the normal stresses should be added both in P^k and in the momentum equations.

The Assignment 1 is mandatory. Do one/many/all of Assignment 2a–2d.

AE.3 DNS Data bases for validating NN model

First a precursor DNS of flow in a half-channel was carried out at $Re_\tau = 550$ ($Re_b = 10\,000$). The grid has $600 \times 150 \times 300$ cell with $0.2 < \Delta y^+ < 5$, $\Delta z^+ = 3$, $\Delta x^+_{in} = 6$. The grids for the hill, wave, contraction and expansion below has $300 \times 300 \times$ for which the mesh at the inlet is identical to that of the channel flow. The boundary conditions for these four flows are:

- Left boundary (low x): inlet b.c. taken from the pre-cursor channel flow.
- Right boundary (high x): outlet
- Low boundary (low y): wall
- High boundary (High y): slip wall, i.e. symmetry b.c.