Stručné shrnutí:

Testované modely:

- Neuronová síť typu MLP – proveden tunning hyperparametrů modelu
- Model XGBOOST – velmi účinný model založený na náhodných lesech, proveden tunning parametrů modelu
- Model SVR – regrese pomocí podpůrných vektorů s RBF jádrem s velmi lehkým tunningem hyperparametrů. Pro každou komponentu [A1,A2,T1,T2] byl udělán vlastní model.

Výsledky na simulovaných datech:

Modely byly porovnány na těch samých datech. Data byla rozdělena na trénovací (90%) a testovací (10%). Do modelu vstupují vždy škálovaná data, tak aby rozsah všech komponent [A1,A2,T1,T2] byl vždy v intervalu <0,1>. Pro model MLP bylo před škálováním [A1,A2,T1,T2] provedeno jejich logaritmování, model pak dává lepší výsledky.

Všechny modely poskytují poměrně dobré výsledky na simulovaných datech.
Koeficient determinace R^2:

Model MLP	**Train R^2**	**Test R^2**
A1	0.9610176368422175	0.9601888333496622
A2	0.8682249426189971	0.8726183922762437
T1	0.9468982712147127	0.9505765718913753
T2	0.9738205103870379	0.9717831761784667

Model XGBOOST	**Train R^2**	**Test R^2**
A1	0.9936155845732516	0.9667369415147182
A2	0.983416610427149	0.8538318291707729
T1	0.9901416401548032	0.9503909874617814
T2	0.9971428027251518	0.9802837853351065

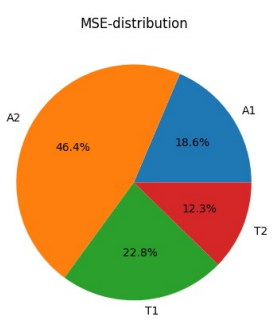
Model SVR	**Train R^2**	**Test R^2**
A1	0.957969654872773	0.957142726381294
A2	0.8460921624708152	0.840520980582669
T1	0.9342935363917684	0.9367389753518843
T2	0.9641186036352347	0.9631837704481786

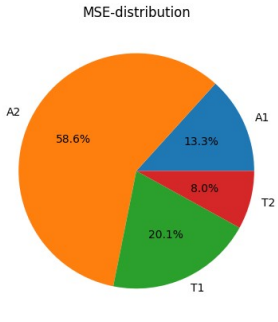
Na testovacích datech vychází MLP srovnatelně s XGBOOST. Model SVR je trochu horší. Model XGBOOST je mírně přeučený, je to vidět na poklesu R2 mezi trénovací a testovací množinou, zejména v A2. Porovnání modelů z hlediska **MSE** na škálovaných testovacích datech, viz následující tabulka, vychází MLP mírně lépe, nejhůře vychází SVR. Opět je vidět mírná přeučenost XGBOOST (lépe se chová na trénovacích, než testovacích datech).

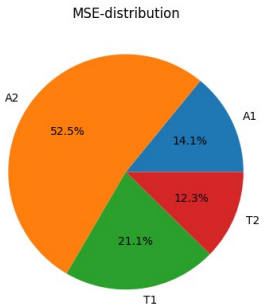
	Train MSE	Test MSE
Model MLP	0.003261030762546111	0.003235847515699445
Model XGBOOST	0.000739664358195811	0.005155383854920913
Model SVR	0.0061656371752120505	0.006272083616339165

Vyhodnocoval jsem i korelovanost predikce modelu s reálnou hodnotou po složkách. Tabulky neuvádím, ale modu dodat data. Korelační koeficient vychází někde mezi <0.92,0.99>, s poklesem v komponentě A2 pro všechny modely.

Z hlediska distribuce chyb mezi komponenty je vidět, že nejvíce se projevuje složka A2 u všech modelů.







Lze říci, že na umělých datech se modely jeví funkčně.

Modely na neznámém signálu:

Vstupní signál: [41.79857899, 47.80665522, 44.95921279, 43.75694528, 40.0470282, 35.33130118, 1260.492509, 594.2514374, 441.3921548, 418.7659755, 402.7798095, 358.1815222, 336.9068776, 321.1603479, 287.4529931, 266.1460777, 240.1359104, 222.0845567, 197.2556906, 156.0158626, **141.8509294**]

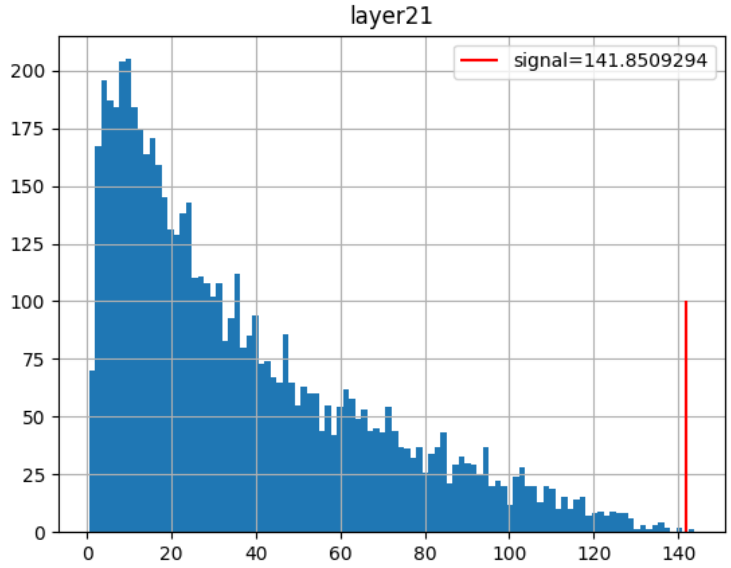
Vstupní signál jsem přeškáloval na interval <0,1> a v případě MLP jsem před škálováním ještě zlogaritmoval [A1,A2,T1,T2]. Provedl predikci modelem a odškáloval na původní hodnoty a případně se zbavil logaritmu pomocí exponenciály. Výsledky jsou shrnuty v následující tabulce:

	A1	A2	T1	T2
MLP	1.3086969e+12	1.9466536e+09	1.8702956e-08	3.0300536e+00
XGBOOST	8.707330e+09	4.531976e+08	-7.845777e-02	4.481385e+00
SVR	1.88432506e+10	-3.33376577e+09	-5.02186165e+00	-4.25631504e+00

Zdá se, že přístup nebyl zcela úspěšný, ale nemám definován pro jaké hodnoty to bude úspěch.

- Vzniklé záporné hodnoty asi půjdou odstranit logaritmování a exponenciála, viz MLP. Chápu to tak, že např. záporné hodnoty mi říkají, že T2 (po úpravě logaritmováním) bude malé.
- Musím udělat hlubší analýzu proč jsou modely odlišné, např komponenta A1 dost skáče. Zatím jsem si všiml, že data co jsem použil pro učení modelu a vstupní signál si velikostně odpovídají až na složku **layer21**, která je ve vstupním signálu větší než v učební množině,

viz histogram hodnot níže. Cvičně jsem si u MLP zmenšoval hodnotu někam k průměru (mean(layer21)=38) a mělo to vliv na pokles komponenty A1. Asi dochází k tomu, že model se nebyl schopen vyrovnat s touto odlehlou hodnotou.



Co dále:

Nevím zda to považujete za perspektivní?

- Pokud ano, tak můžeme nagenerovat umělá data, kde bude větší rozsah layer21 a já modely znovu naučím a uvidíme.
- Analýza proč jsou modely odlišné? Případně změnit preprocesing parametrů, tak aby to bylo stabilnější.
- Mohu vyzkoušet konvoluční sítě, ale to si musím trochu rozmyslet.