# Annex 3.0

Data for our third iteration of ML algorithms. Removed 0 and 10

# Annex 3.1

## NR

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'svd'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'friedman\_mse', 'decisiontreeregressor\_\_max\_features': 2, 'decisiontreeregressor\_\_min\_samples\_split': 5, 'decisiontreeregressor\_\_splitter': 'best'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.01, 'gradientboostingregressor\_\_loss': 'squared\_error', 'gradientboostingregressor\_\_n\_estimators': 100, 'gradientboostingregressor\_\_warm\_start': True}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'friedman\_mse', 'randomforestregressor\_\_max\_features': 'sqrt', 'randomforestregressor\_\_min\_samples\_split': 5, 'randomforestregressor\_\_n\_estimators': 100}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 0.001, 'adaboostregressor\_\_loss': 'exponential', 'adaboostregressor\_\_n\_estimators': 20}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'ball\_tree', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cityblock', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'uniform'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'tanh', 'mlpregressor\_\_hidden\_layer\_sizes': (100, 100, 100, 100), 'mlpregressor\_\_learning\_rate': 'adaptive', 'mlpregressor\_\_solver': 'sgd'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': False, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.25, 'elasticnet\_\_positive': False, 'elasticnet\_\_precompute': False, 'elasticnet\_\_selection': 'random', 'elasticnet\_\_warm\_start': True}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'constant', 'sgdregressor\_\_loss': 'squared\_error', 'sgdregressor\_\_penalty': 'elasticnet', 'sgdregressor\_\_warm\_start': False}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 'scale', 'svr\_\_kernel': 'rbf', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-05, 'bayesianridge\_\_alpha\_2': 1e-05, 'bayesianridge\_\_lambda\_1': 1e-07, 'bayesianridge\_\_lambda\_2': 1e-05}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': True}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'absolute\_error', 'ransacregressor\_\_max\_trials': 50, 'ransacregressor\_\_min\_samples': 50}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 10000, 'theilsenregressor\_\_n\_subsamples': None}

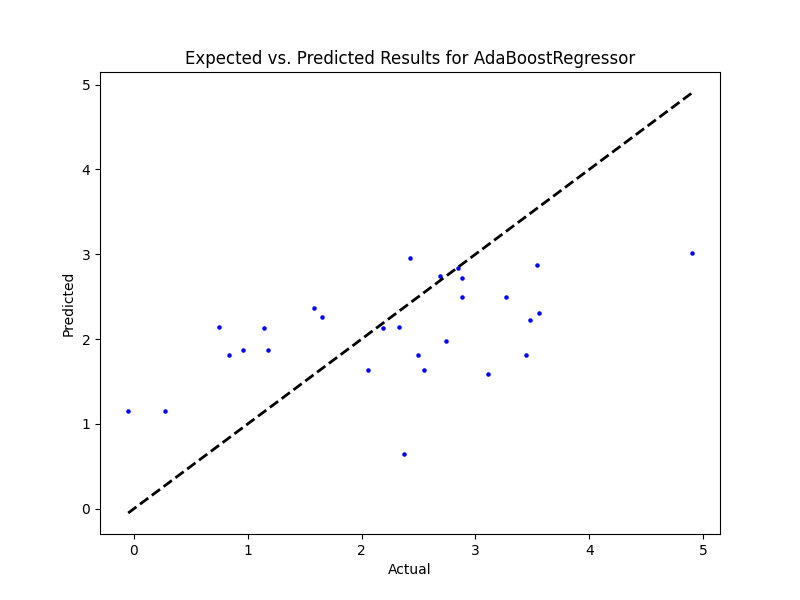
**TensorFlow()**

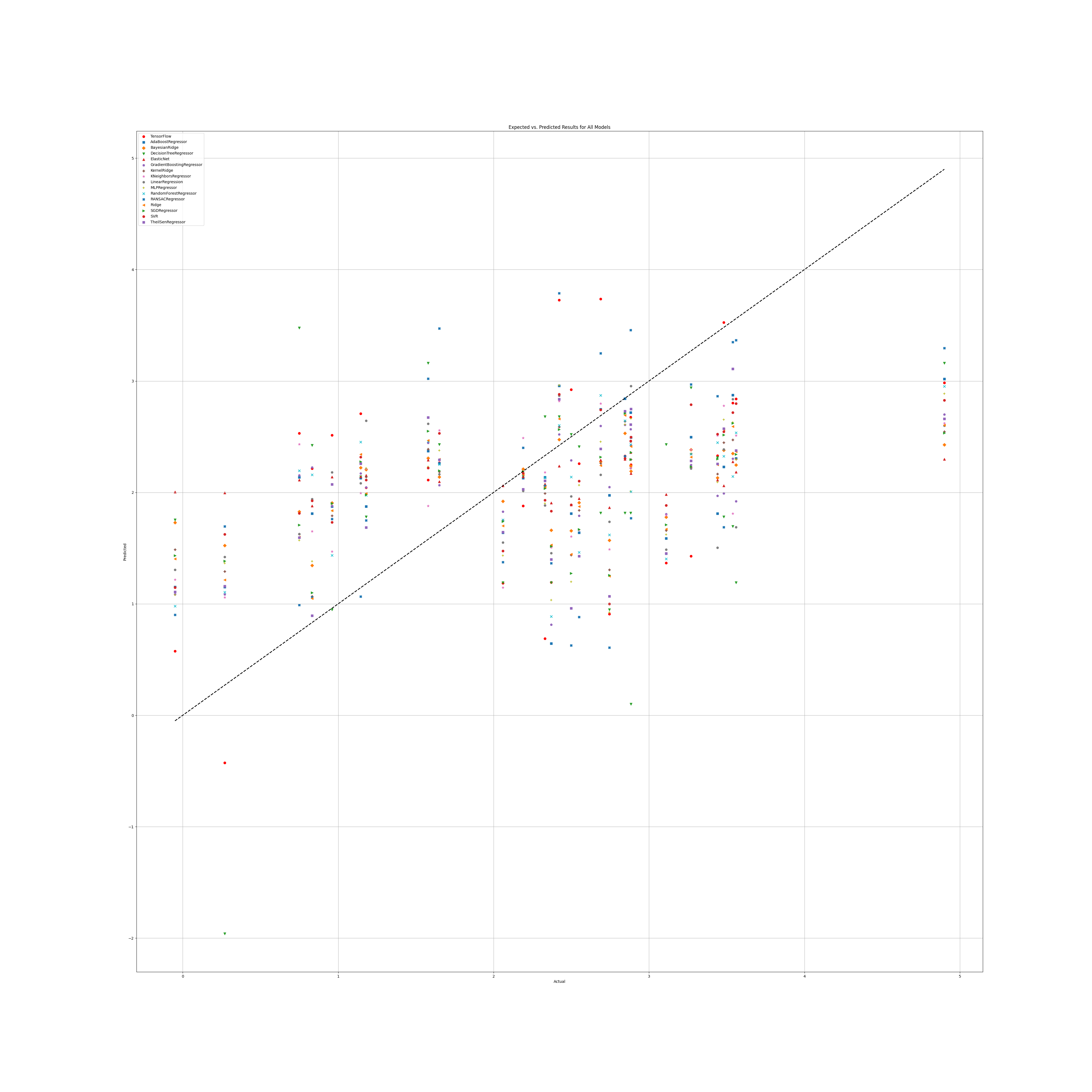
TensorFlow RMSE: 1.1449841419442457

### Prediction results

Model: Ridge  
Model saved as NR\_filtered\Ridge\_model.pkl  
RMSE: 0.9694131214972044  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.603560862099511  
Sample 2: Real NR = 1.65, Predicted NR = 2.29707995689129  
Sample 3: Real NR = 2.33, Predicted NR = 2.0447674409382053  
Sample 4: Real NR = 0.27, Predicted NR = 1.2159828707228502  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.161458669714283  
  
  
Model: DecisionTreeRegressor  
Model saved as NR\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 1.410591033742122  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 3.1607314641791024  
Sample 2: Real NR = 1.65, Predicted NR = 2.4303669461295807  
Sample 3: Real NR = 2.33, Predicted NR = 2.68  
Sample 4: Real NR = 0.27, Predicted NR = -1.96  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.140914207819342  
  
  
Model: GradientBoostingRegressor  
Model saved as NR\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 1.031134894012954  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.6994584193947575  
Sample 2: Real NR = 1.65, Predicted NR = 2.0659021661611696  
Sample 3: Real NR = 2.33, Predicted NR = 1.8830337593267814  
Sample 4: Real NR = 0.27, Predicted NR = 1.08519935842566  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.169553788258012  
  
  
Model: RandomForestRegressor  
Model saved as NR\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 1.003224359794718  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.9515055856721166  
Sample 2: Real NR = 1.65, Predicted NR = 2.2506189937257797  
Sample 3: Real NR = 2.33, Predicted NR = 2.130876359711817  
Sample 4: Real NR = 0.27, Predicted NR = 1.1048308551872343  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.1615158661863108  
  
  
Model: AdaBoostRegressor  
Model saved as NR\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 0.9751884611107077  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 3.0192176969108497  
Sample 2: Real NR = 1.65, Predicted NR = 2.262564642563506  
Sample 3: Real NR = 2.33, Predicted NR = 2.136693193830744  
Sample 4: Real NR = 0.27, Predicted NR = 1.1512057170124281  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.128637857478825  
  
  
Model: KNeighborsRegressor  
Model saved as NR\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 0.9901185167237185  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.6225274626661634  
Sample 2: Real NR = 1.65, Predicted NR = 2.5583985227217307  
Sample 3: Real NR = 2.33, Predicted NR = 2.18  
Sample 4: Real NR = 0.27, Predicted NR = 1.059047659869151  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.4889424677702285  
  
  
Model: MLPRegressor  
Model saved as NR\_filtered\MLPRegressor\_model.pkl  
RMSE: 1.0085574065444884  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.886392253236781  
Sample 2: Real NR = 1.65, Predicted NR = 2.376315784692549  
Sample 3: Real NR = 2.33, Predicted NR = 1.9011835694124013  
Sample 4: Real NR = 0.27, Predicted NR = 1.363726843224886  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.1642488247931153  
  
  
Model: ElasticNet  
Model saved as NR\_filtered\ElasticNet\_model.pkl  
RMSE: 1.0961267600562088  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.2972427865213767  
Sample 2: Real NR = 1.65, Predicted NR = 2.096073664369582  
Sample 3: Real NR = 2.33, Predicted NR = 2.074591466475925  
Sample 4: Real NR = 0.27, Predicted NR = 1.9966549692073352  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.1423694747604918  
  
  
Model: SGDRegressor  
Model saved as NR\_filtered\SGDRegressor\_model.pkl  
RMSE: 0.9975459728624405  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.5333802919439927  
Sample 2: Real NR = 1.65, Predicted NR = 2.1946950978795545  
Sample 3: Real NR = 2.33, Predicted NR = 2.0348807659490613  
Sample 4: Real NR = 0.27, Predicted NR = 1.383456147014491  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.186783622401383  
  
  
Model: SVR  
Model saved as NR\_filtered\SVR\_model.pkl  
RMSE: 0.9231177739445382  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.828185737069092  
Sample 2: Real NR = 1.65, Predicted NR = 2.5305588108921615  
Sample 3: Real NR = 2.33, Predicted NR = 1.9295550277783868  
Sample 4: Real NR = 0.27, Predicted NR = 1.6254295561687095  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.1735244403734053  
  
  
Model: BayesianRidge  
Model saved as NR\_filtered\BayesianRidge\_model.pkl  
RMSE: 1.018415836405853  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.426779148957112  
Sample 2: Real NR = 1.65, Predicted NR = 2.138619479187248  
Sample 3: Real NR = 2.33, Predicted NR = 2.06025581500104  
Sample 4: Real NR = 0.27, Predicted NR = 1.5231430441200042  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.2093627971413925  
  
  
Model: KernelRidge  
Model saved as NR\_filtered\KernelRidge\_model.pkl  
RMSE: 0.9932695105613597  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.545439841243182  
Sample 2: Real NR = 1.65, Predicted NR = 2.1849858994651594  
Sample 3: Real NR = 2.33, Predicted NR = 1.9922311021062926  
Sample 4: Real NR = 0.27, Predicted NR = 1.2919337377900681  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.1624255738679166  
  
  
Model: LinearRegression  
Model saved as NR\_filtered\LinearRegression\_model.pkl  
RMSE: 1.0773770146590942  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.598438962791487  
Sample 2: Real NR = 1.65, Predicted NR = 2.162452263690371  
Sample 3: Real NR = 2.33, Predicted NR = 1.8829510096716198  
Sample 4: Real NR = 0.27, Predicted NR = 1.4205886315284468  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.0140315430331137  
  
  
Model: RANSACRegressor  
Model saved as NR\_filtered\RANSACRegressor\_model.pkl  
RMSE: 1.1165365664480333  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 3.296333733706166  
Sample 2: Real NR = 1.65, Predicted NR = 3.4710391253271964  
Sample 3: Real NR = 2.33, Predicted NR = 2.0604380101028377  
Sample 4: Real NR = 0.27, Predicted NR = 1.694339736371663  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.400355283410199  
  
  
Model: TheilSenRegressor  
Model saved as NR\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 0.9950386373121319  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = 2.661813486467137  
Sample 2: Real NR = 1.65, Predicted NR = 2.2932375011274515  
Sample 3: Real NR = 2.33, Predicted NR = 2.1051996192972253  
Sample 4: Real NR = 0.27, Predicted NR = 1.1584901385996147  
Sample 5: Real NR = 2.189940075290342, Predicted NR = 2.029824317181642  
  
  
Model: TensorFlow  
Model saved as NR\_filtered\TensorFlow\_model.h5  
RMSE: 1.1449841419442457  
Sample predictions:  
Sample 1: Real NR = 4.9, Predicted NR = [2.9843333]  
Sample 2: Real NR = 1.65, Predicted NR = [2.2660177]  
Sample 3: Real NR = 2.33, Predicted NR = [0.68768454]  
Sample 4: Real NR = 0.27, Predicted NR = [-0.42583457]  
Sample 5: Real NR = 2.189940075290342, Predicted NR = [1.880234]

### Graph results





## NR Benefit

### Grid search results

Ridge()  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'saga'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'friedman\_mse', 'decisiontreeregressor\_\_max\_features': 'log2', 'decisiontreeregressor\_\_min\_samples\_split': 4, 'decisiontreeregressor\_\_splitter': 'random'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.001, 'gradientboostingregressor\_\_loss': 'squared\_error', 'gradientboostingregressor\_\_n\_estimators': 100, 'gradientboostingregressor\_\_warm\_start': False}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'friedman\_mse', 'randomforestregressor\_\_max\_features': 3, 'randomforestregressor\_\_min\_samples\_split': 5, 'randomforestregressor\_\_n\_estimators': 50}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 0.01, 'adaboostregressor\_\_loss': 'exponential', 'adaboostregressor\_\_n\_estimators': 50}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'ball\_tree', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cityblock', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'distance'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'logistic', 'mlpregressor\_\_hidden\_layer\_sizes': (100, 100, 100, 100), 'mlpregressor\_\_learning\_rate': 'constant', 'mlpregressor\_\_solver': 'lbfgs'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': True, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.5, 'elasticnet\_\_positive': True, 'elasticnet\_\_precompute': True, 'elasticnet\_\_selection': 'cyclic', 'elasticnet\_\_warm\_start': True}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'adaptive', 'sgdregressor\_\_loss': 'huber', 'sgdregressor\_\_penalty': 'l1', 'sgdregressor\_\_warm\_start': False}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 1.0, 'svr\_\_kernel': 'rbf', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-07, 'bayesianridge\_\_alpha\_2': 1e-05, 'bayesianridge\_\_lambda\_1': 1e-05, 'bayesianridge\_\_lambda\_2': 1e-07}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': True}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'absolute\_error', 'ransacregressor\_\_max\_trials': 10, 'ransacregressor\_\_min\_samples': 1}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 10, 'theilsenregressor\_\_n\_subsamples': 25}

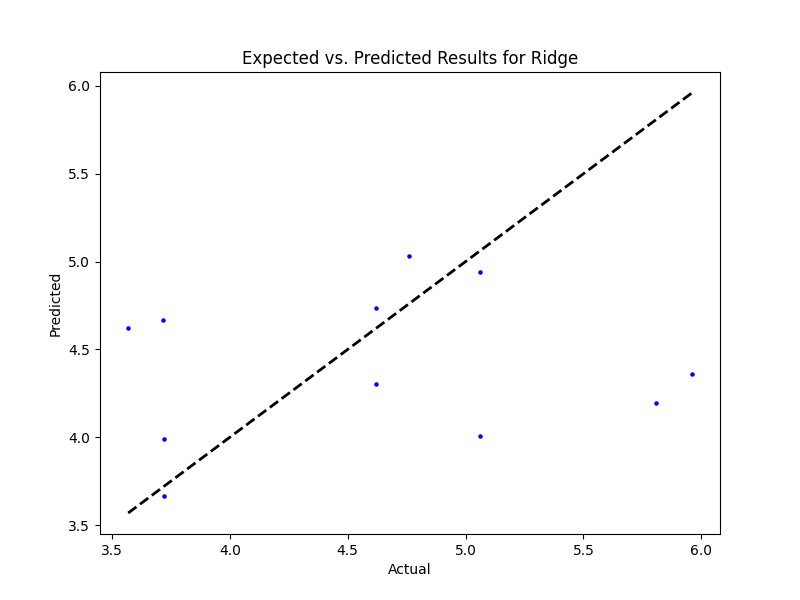
**TensorFlow()**

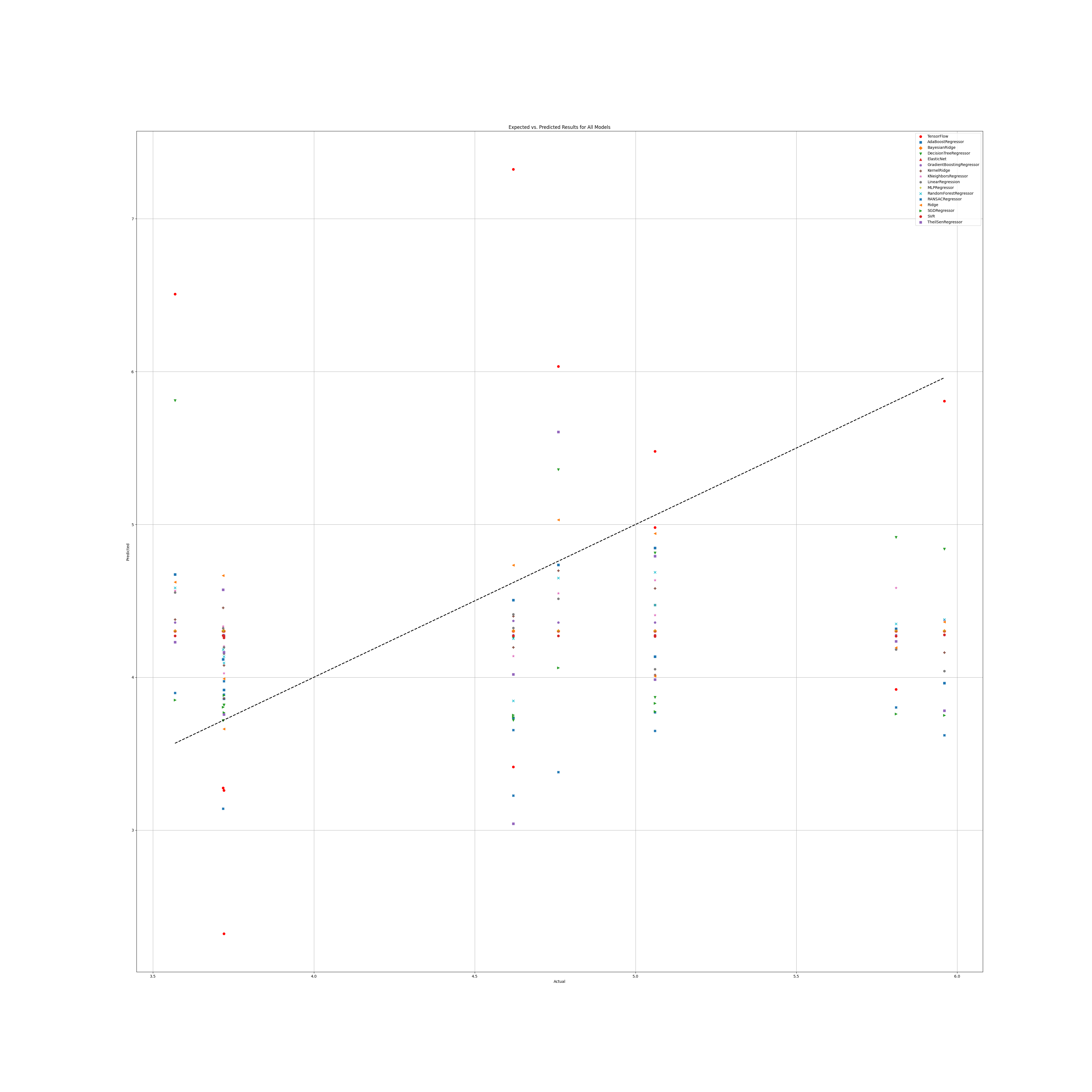
TensorFlow RMSE: 1.5124679338430265

### Prediction results

Model: Ridge  
Model saved as NR\_benefit\_filtered\Ridge\_model.pkl  
RMSE: 0.8821999755503492  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.006911283204991  
Sample 2: Real NR = 5.81, Predicted NR = 4.19398097592858  
Sample 3: Real NR = 4.62, Predicted NR = 4.734729480080716  
Sample 4: Real NR = 3.72, Predicted NR = 3.9917273219733787  
Sample 5: Real NR = 5.06, Predicted NR = 4.940626182665007  
  
  
Model: DecisionTreeRegressor  
Model saved as NR\_benefit\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 0.9479333726449211  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 3.8689743589743584  
Sample 2: Real NR = 5.81, Predicted NR = 4.916666666666667  
Sample 3: Real NR = 4.62, Predicted NR = 4.2599430199430195  
Sample 4: Real NR = 3.72, Predicted NR = 3.8200000000000003  
Sample 5: Real NR = 5.06, Predicted NR = 4.814985754985755  
  
  
Model: GradientBoostingRegressor  
Model saved as NR\_benefit\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 0.8626582613426758  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.277131675116871  
Sample 2: Real NR = 5.81, Predicted NR = 4.277131675116871  
Sample 3: Real NR = 4.62, Predicted NR = 4.370034685593  
Sample 4: Real NR = 3.72, Predicted NR = 4.277131675116871  
Sample 5: Real NR = 5.06, Predicted NR = 4.358118024507926  
  
  
Model: RandomForestRegressor  
Model saved as NR\_benefit\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 0.8216318143669734  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.473627738434405  
Sample 2: Real NR = 5.81, Predicted NR = 4.350215811965812  
Sample 3: Real NR = 4.62, Predicted NR = 4.2537098249898255  
Sample 4: Real NR = 3.72, Predicted NR = 4.136248074888075  
Sample 5: Real NR = 5.06, Predicted NR = 4.687626336996337  
  
  
Model: AdaBoostRegressor  
Model saved as NR\_benefit\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 0.9219710371296943  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.135244444444444  
Sample 2: Real NR = 5.81, Predicted NR = 4.318664529914531  
Sample 3: Real NR = 4.62, Predicted NR = 4.505  
Sample 4: Real NR = 3.72, Predicted NR = 3.9184696784696778  
Sample 5: Real NR = 5.06, Predicted NR = 4.846470588235294  
  
  
Model: KNeighborsRegressor  
Model saved as NR\_benefit\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 0.7801819001566108  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.407146536074741  
Sample 2: Real NR = 5.81, Predicted NR = 4.585532249773264  
Sample 3: Real NR = 4.62, Predicted NR = 4.13957461517849  
Sample 4: Real NR = 3.72, Predicted NR = 4.203104348990253  
Sample 5: Real NR = 5.06, Predicted NR = 4.636627162008076  
  
  
Model: MLPRegressor  
Model saved as NR\_benefit\_filtered\MLPRegressor\_model.pkl  
RMSE: 0.8597334586548661  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.303786107241497  
Sample 2: Real NR = 5.81, Predicted NR = 4.303965566693818  
Sample 3: Real NR = 4.62, Predicted NR = 4.303168317125757  
Sample 4: Real NR = 3.72, Predicted NR = 4.304034143723836  
Sample 5: Real NR = 5.06, Predicted NR = 4.3033698747356715  
  
  
Model: ElasticNet  
Model saved as NR\_benefit\_filtered\ElasticNet\_model.pkl  
RMSE: 0.8597269578437028  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.303598082134667  
Sample 2: Real NR = 5.81, Predicted NR = 4.303598082134667  
Sample 3: Real NR = 4.62, Predicted NR = 4.303598082134667  
Sample 4: Real NR = 3.72, Predicted NR = 4.303598082134667  
Sample 5: Real NR = 5.06, Predicted NR = 4.303598082134667  
  
  
Model: SGDRegressor  
Model saved as NR\_benefit\_filtered\SGDRegressor\_model.pkl  
RMSE: 1.142871990333712  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 3.776325440863272  
Sample 2: Real NR = 5.81, Predicted NR = 3.7603188894862316  
Sample 3: Real NR = 4.62, Predicted NR = 3.740818930925844  
Sample 4: Real NR = 3.72, Predicted NR = 3.881491535127634  
Sample 5: Real NR = 5.06, Predicted NR = 3.829495438060118  
  
  
Model: SVR  
Model saved as NR\_benefit\_filtered\SVR\_model.pkl  
RMSE: 0.8697415541120697  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.268176409384709  
Sample 2: Real NR = 5.81, Predicted NR = 4.2699854189547075  
Sample 3: Real NR = 4.62, Predicted NR = 4.2713233036532685  
Sample 4: Real NR = 3.72, Predicted NR = 4.2592499648818425  
Sample 5: Real NR = 5.06, Predicted NR = 4.2724536020975155  
  
  
Model: BayesianRidge  
Model saved as NR\_benefit\_filtered\BayesianRidge\_model.pkl  
RMSE: 0.859688954193672  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.303204513596137  
Sample 2: Real NR = 5.81, Predicted NR = 4.303746681098199  
Sample 3: Real NR = 4.62, Predicted NR = 4.303646955187433  
Sample 4: Real NR = 3.72, Predicted NR = 4.303443110735751  
Sample 5: Real NR = 5.06, Predicted NR = 4.304193307314215  
  
  
Model: KernelRidge  
Model saved as NR\_benefit\_filtered\KernelRidge\_model.pkl  
RMSE: 0.8927292234013787  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.015716428872498  
Sample 2: Real NR = 5.81, Predicted NR = 4.186131412859917  
Sample 3: Real NR = 4.62, Predicted NR = 4.39929908517979  
Sample 4: Real NR = 3.72, Predicted NR = 4.078698689351465  
Sample 5: Real NR = 5.06, Predicted NR = 4.582302240978141  
  
  
Model: LinearRegression  
Model saved as NR\_benefit\_filtered\LinearRegression\_model.pkl  
RMSE: 0.9361814048264211  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 4.054225566183622  
Sample 2: Real NR = 5.81, Predicted NR = 4.181444769312841  
Sample 3: Real NR = 4.62, Predicted NR = 4.413262101073912  
Sample 4: Real NR = 3.72, Predicted NR = 4.155671637194567  
Sample 5: Real NR = 5.06, Predicted NR = 4.472517081594321  
  
  
Model: RANSACRegressor  
Model saved as NR\_benefit\_filtered\RANSACRegressor\_model.pkl  
RMSE: 1.2953017435620287  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 3.772162466854903  
Sample 2: Real NR = 5.81, Predicted NR = 3.803400833815805  
Sample 3: Real NR = 4.62, Predicted NR = 3.225765363402931  
Sample 4: Real NR = 3.72, Predicted NR = 3.9754469577296763  
Sample 5: Real NR = 5.06, Predicted NR = 3.6497245522869526  
  
  
Model: TheilSenRegressor  
Model saved as NR\_benefit\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 1.102865302828997  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = 3.9849964138308724  
Sample 2: Real NR = 5.81, Predicted NR = 4.235996284091311  
Sample 3: Real NR = 4.62, Predicted NR = 3.0427820956334575  
Sample 4: Real NR = 3.72, Predicted NR = 4.163524408786949  
Sample 5: Real NR = 5.06, Predicted NR = 4.792842353923045  
  
  
Model: TensorFlow  
Model saved as NR\_benefit\_filtered\TensorFlow\_model.h5  
RMSE: 1.5124679338430265  
Sample predictions:  
Sample 1: Real NR = 5.06, Predicted NR = [4.9799924]  
Sample 2: Real NR = 5.81, Predicted NR = [3.920893]  
Sample 3: Real NR = 4.62, Predicted NR = [7.3241687]  
Sample 4: Real NR = 3.72, Predicted NR = [3.2604036]  
Sample 5: Real NR = 5.06, Predicted NR = [5.47834]

### Graph results





# Annex 3.2

## WS

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'lsqr'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'squared\_error', 'decisiontreeregressor\_\_max\_features': 'log2', 'decisiontreeregressor\_\_min\_samples\_split': 5, 'decisiontreeregressor\_\_splitter': 'best'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.01, 'gradientboostingregressor\_\_loss': 'huber', 'gradientboostingregressor\_\_n\_estimators': 250, 'gradientboostingregressor\_\_warm\_start': True}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'squared\_error', 'randomforestregressor\_\_max\_features': 'sqrt', 'randomforestregressor\_\_min\_samples\_split': 2, 'randomforestregressor\_\_n\_estimators': 100}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 0.1, 'adaboostregressor\_\_loss': 'exponential', 'adaboostregressor\_\_n\_estimators': 100}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'ball\_tree', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cityblock', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'distance'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'relu', 'mlpregressor\_\_hidden\_layer\_sizes': (50, 50, 50), 'mlpregressor\_\_learning\_rate': 'adaptive', 'mlpregressor\_\_solver': 'sgd'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': True, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.25, 'elasticnet\_\_positive': False, 'elasticnet\_\_precompute': True, 'elasticnet\_\_selection': 'random', 'elasticnet\_\_warm\_start': True}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'constant', 'sgdregressor\_\_loss': 'squared\_error', 'sgdregressor\_\_penalty': 'elasticnet', 'sgdregressor\_\_warm\_start': False}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 'scale', 'svr\_\_kernel': 'rbf', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-07, 'bayesianridge\_\_alpha\_2': 1e-05, 'bayesianridge\_\_lambda\_1': 1e-05, 'bayesianridge\_\_lambda\_2': 1e-07}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 0.5, 'kernelridge\_\_degree': 3, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': False}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'squared\_error', 'ransacregressor\_\_max\_trials': 150, 'ransacregressor\_\_min\_samples': 2}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 1000, 'theilsenregressor\_\_n\_subsamples': None}

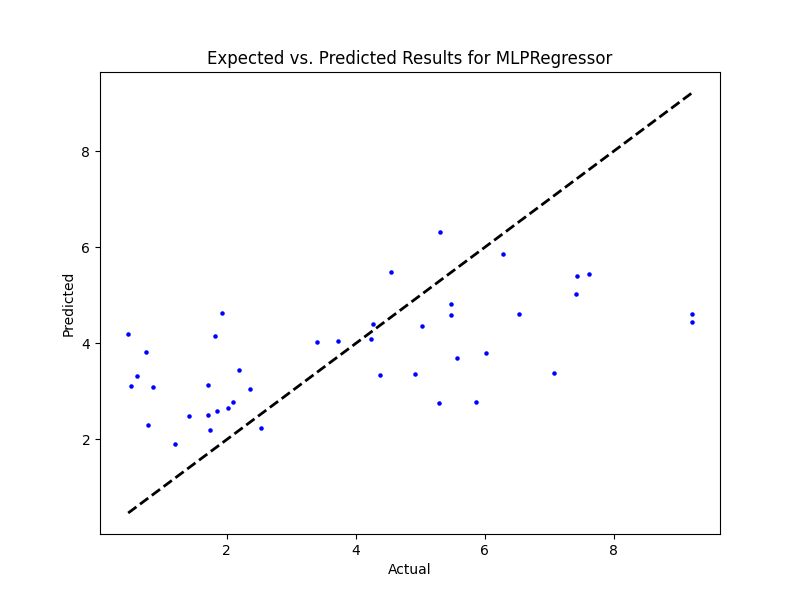
**TensorFlow()**

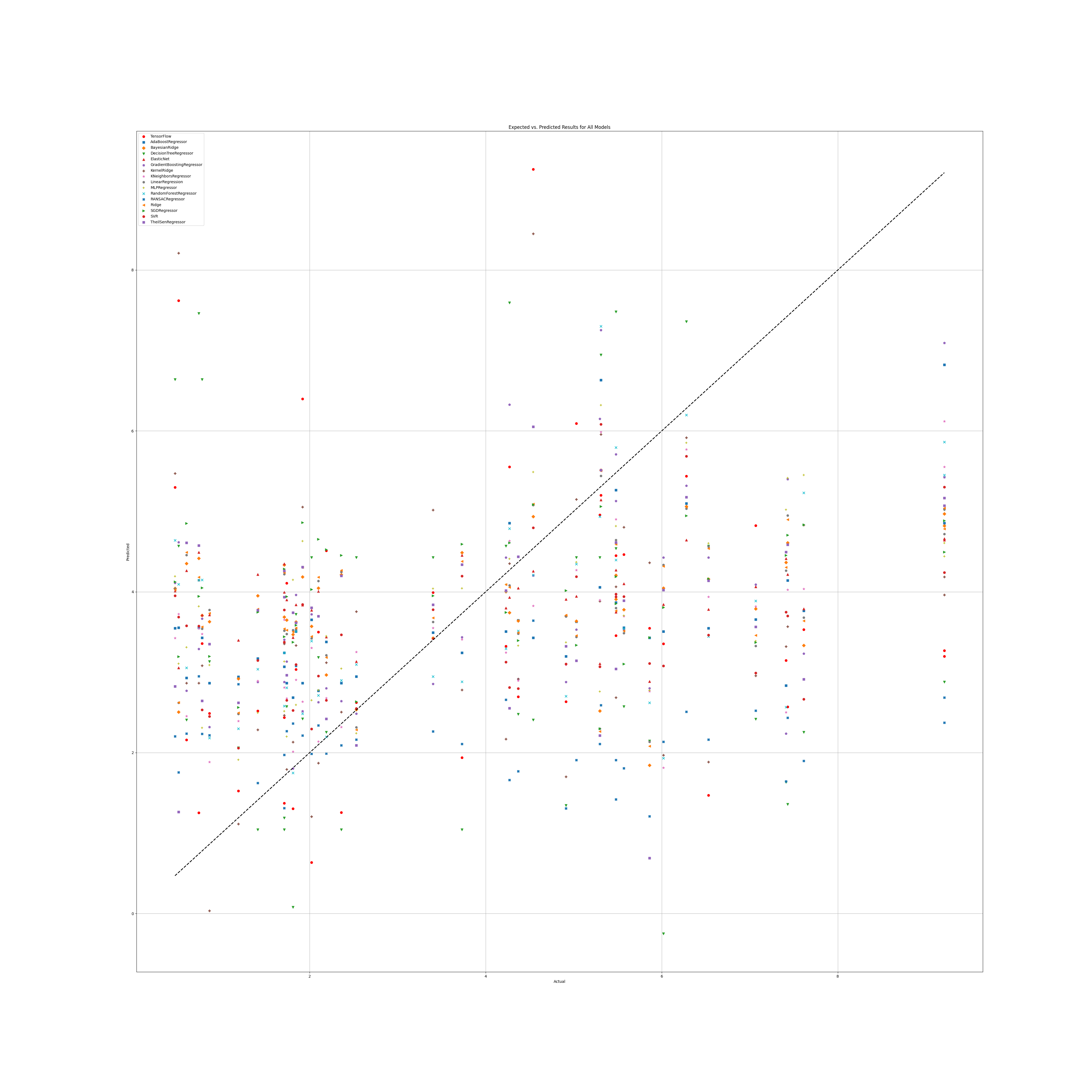
TensorFlow RMSE: 2.8585533239459195

### Prediction results

Model: Ridge  
Model saved as WS\_filtered\Ridge\_model.pkl  
RMSE: 2.3406599232017165  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 2.6232481533073293  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.090307129367414  
Sample 3: Real NR = 5.31, Predicted NR = 5.50571272365558  
Sample 4: Real NR = 1.41, Predicted NR = 3.785524807741328  
Sample 5: Real NR = 5.86, Predicted NR = 2.0829214515377696  
  
  
Model: DecisionTreeRegressor  
Model saved as WS\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 3.302728292591861  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 4.570245623941275  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.570245623941275  
Sample 3: Real NR = 5.31, Predicted NR = 6.9433333333333325  
Sample 4: Real NR = 1.41, Predicted NR = 1.0442772444946358  
Sample 5: Real NR = 5.86, Predicted NR = 2.146993224167137  
  
  
Model: GradientBoostingRegressor  
Model saved as WS\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 2.1121986699095507  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 4.617203323443564  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.426634131277063  
Sample 3: Real NR = 5.31, Predicted NR = 7.25324194326474  
Sample 4: Real NR = 1.41, Predicted NR = 2.877326516079142  
Sample 5: Real NR = 5.86, Predicted NR = 2.799503097027023  
  
  
Model: RandomForestRegressor  
Model saved as WS\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 2.193833115701528  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 4.092872953133823  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 3.2931409373235447  
Sample 3: Real NR = 5.31, Predicted NR = 7.299557933370977  
Sample 4: Real NR = 1.41, Predicted NR = 3.0370499153020893  
Sample 5: Real NR = 5.86, Predicted NR = 2.622661603613777  
  
  
Model: AdaBoostRegressor  
Model saved as WS\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 2.111752507319311  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 3.55330568887634  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 3.5082462886590577  
Sample 3: Real NR = 5.31, Predicted NR = 6.632400875900346  
Sample 4: Real NR = 1.41, Predicted NR = 3.1716765608069974  
Sample 5: Real NR = 5.86, Predicted NR = 3.4299828037575093  
  
  
Model: KNeighborsRegressor  
Model saved as WS\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 2.0981518556824144  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 3.722291307941659  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 3.2452849179003063  
Sample 3: Real NR = 5.31, Predicted NR = 5.989083940927828  
Sample 4: Real NR = 1.41, Predicted NR = 2.891970169201925  
Sample 5: Real NR = 5.86, Predicted NR = 2.7637997391364912  
  
  
Model: MLPRegressor  
Model saved as WS\_filtered\MLPRegressor\_model.pkl  
RMSE: 2.031859236223153  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 3.106474221031183  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.090992164261051  
Sample 3: Real NR = 5.31, Predicted NR = 6.320559149019052  
Sample 4: Real NR = 1.41, Predicted NR = 2.494387345023199  
Sample 5: Real NR = 5.86, Predicted NR = 2.7751856463091036  
  
  
Model: ElasticNet  
Model saved as WS\_filtered\ElasticNet\_model.pkl  
RMSE: 2.382263573548347  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 3.056343158699799  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 3.7991282298139444  
Sample 3: Real NR = 5.31, Predicted NR = 5.142939675448469  
Sample 4: Real NR = 1.41, Predicted NR = 4.216602367851283  
Sample 5: Real NR = 5.86, Predicted NR = 2.886924696145165  
  
  
Model: SGDRegressor  
Model saved as WS\_filtered\SGDRegressor\_model.pkl  
RMSE: 2.409414915893018  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 3.1941449990599593  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 3.7454921322551153  
Sample 3: Real NR = 5.31, Predicted NR = 5.059419357809283  
Sample 4: Real NR = 1.41, Predicted NR = 3.746386741329666  
Sample 5: Real NR = 5.86, Predicted NR = 3.4327357230088653  
  
  
Model: SVR  
Model saved as WS\_filtered\SVR\_model.pkl  
RMSE: 2.2985665432751876  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 3.686687270064108  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 3.126790578062225  
Sample 3: Real NR = 5.31, Predicted NR = 6.081154300565403  
Sample 4: Real NR = 1.41, Predicted NR = 3.147430044558572  
Sample 5: Real NR = 5.86, Predicted NR = 3.109037907436716  
  
  
Model: BayesianRidge  
Model saved as WS\_filtered\BayesianRidge\_model.pkl  
RMSE: 2.372658210914382  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 2.5054461053672457  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.000580821145761  
Sample 3: Real NR = 5.31, Predicted NR = 5.513867741144713  
Sample 4: Real NR = 1.41, Predicted NR = 3.9507860737624254  
Sample 5: Real NR = 5.86, Predicted NR = 1.845563307498617  
  
  
Model: KernelRidge  
Model saved as WS\_filtered\KernelRidge\_model.pkl  
RMSE: 2.7473464537675762  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 8.210214086780702  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 2.1704600365907325  
Sample 3: Real NR = 5.31, Predicted NR = 5.957952421210678  
Sample 4: Real NR = 1.41, Predicted NR = 2.28587368892449  
Sample 5: Real NR = 5.86, Predicted NR = 4.362064253661856  
  
  
Model: LinearRegression  
Model saved as WS\_filtered\LinearRegression\_model.pkl  
RMSE: 2.3381854892139216  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 2.6225662108624057  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.004940757672809  
Sample 3: Real NR = 5.31, Predicted NR = 5.440074446700106  
Sample 4: Real NR = 1.41, Predicted NR = 3.7589308416248564  
Sample 5: Real NR = 5.86, Predicted NR = 2.1343675283184376  
  
  
Model: RANSACRegressor  
Model saved as WS\_filtered\RANSACRegressor\_model.pkl  
RMSE: 3.0742393008922826  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 1.757089504785053  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 2.6578374459371994  
Sample 3: Real NR = 5.31, Predicted NR = 2.5907262950440697  
Sample 4: Real NR = 1.41, Predicted NR = 1.6230440109062867  
Sample 5: Real NR = 5.86, Predicted NR = 1.2094887009275062  
  
  
Model: TheilSenRegressor  
Model saved as WS\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 2.401089323091398  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = 1.262708236420527  
Sample 2: Real NR = 4.229249011857707, Predicted NR = 4.014977933949092  
Sample 3: Real NR = 5.31, Predicted NR = 5.511105667402175  
Sample 4: Real NR = 1.41, Predicted NR = 3.770158170513845  
Sample 5: Real NR = 5.86, Predicted NR = 0.6895573769929917  
  
  
Model: TensorFlow  
Model saved as WS\_filtered\TensorFlow\_model.h5  
RMSE: 2.8585533239459195  
Sample predictions:  
Sample 1: Real NR = 0.51, Predicted NR = [7.619506]  
Sample 2: Real NR = 4.229249011857707, Predicted NR = [3.3225443]  
Sample 3: Real NR = 5.31, Predicted NR = [5.1986003]  
Sample 4: Real NR = 1.41, Predicted NR = [2.5184596]  
Sample 5: Real NR = 5.86, Predicted NR = [3.5481985]

### Graph results





## WS Benefit

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'sag'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'poisson', 'decisiontreeregressor\_\_max\_features': 1, 'decisiontreeregressor\_\_min\_samples\_split': 2, 'decisiontreeregressor\_\_splitter': 'random'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.001, 'gradientboostingregressor\_\_loss': 'squared\_error', 'gradientboostingregressor\_\_n\_estimators': 25, 'gradientboostingregressor\_\_warm\_start': True}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'squared\_error', 'randomforestregressor\_\_max\_features': 'sqrt', 'randomforestregressor\_\_min\_samples\_split': 2, 'randomforestregressor\_\_n\_estimators': 1}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 0.001, 'adaboostregressor\_\_loss': 'linear', 'adaboostregressor\_\_n\_estimators': 1}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'ball\_tree', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'euclidean', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'uniform'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'logistic', 'mlpregressor\_\_hidden\_layer\_sizes': (100, 100, 100, 100), 'mlpregressor\_\_learning\_rate': 'constant', 'mlpregressor\_\_solver': 'lbfgs'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': True, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.5, 'elasticnet\_\_positive': True, 'elasticnet\_\_precompute': True, 'elasticnet\_\_selection': 'cyclic', 'elasticnet\_\_warm\_start': True}  
**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'adaptive', 'sgdregressor\_\_loss': 'huber', 'sgdregressor\_\_penalty': 'elasticnet', 'sgdregressor\_\_warm\_start': False}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 5.0, 'svr\_\_kernel': 'rbf', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-07, 'bayesianridge\_\_alpha\_2': 1e-05, 'bayesianridge\_\_lambda\_1': 1e-05, 'bayesianridge\_\_lambda\_2': 1e-07}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': True}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'absolute\_error', 'ransacregressor\_\_max\_trials': 150, 'ransacregressor\_\_min\_samples': 2}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 10000, 'theilsenregressor\_\_n\_subsamples': None}

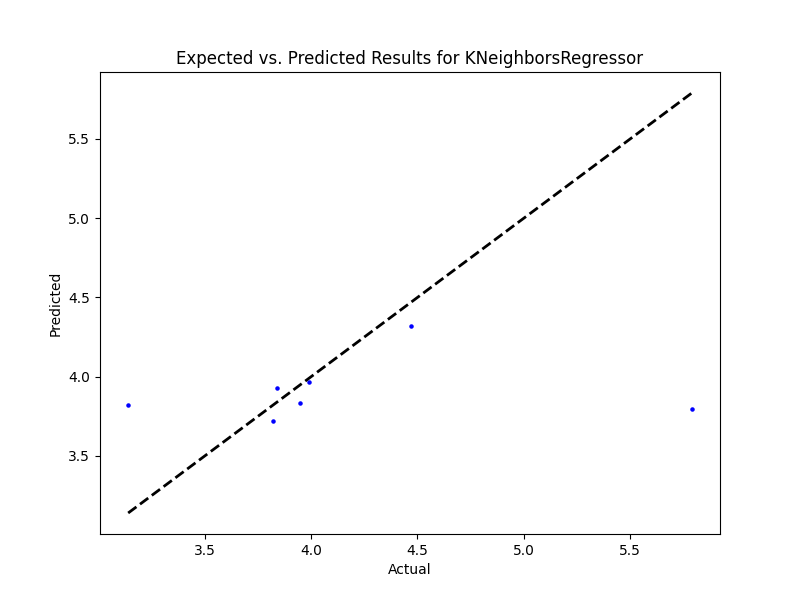
**TensorFlow()**

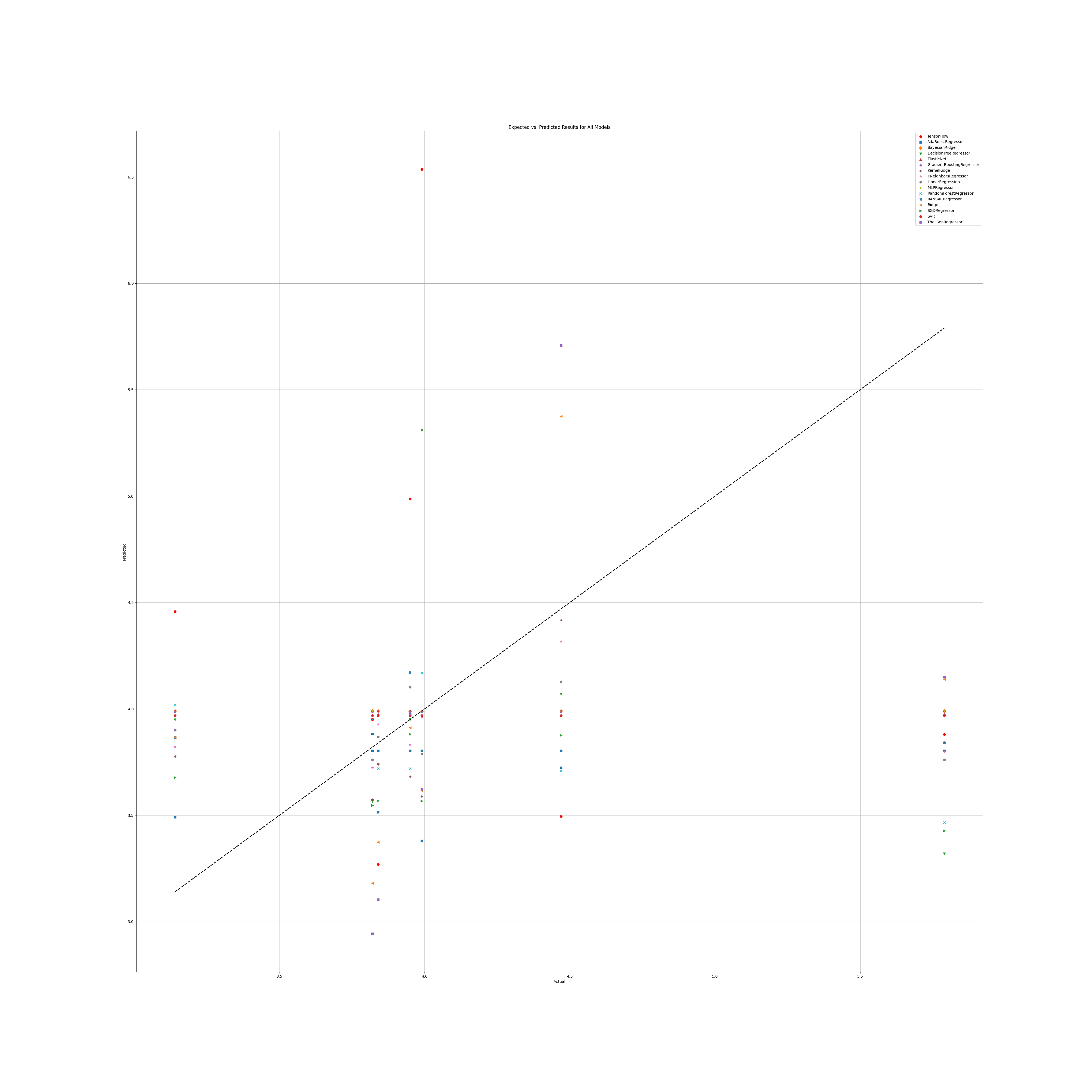
TensorFlow RMSE: 1.4261565749474638

### Prediction results

Model: Ridge  
Model saved as WS\_benefit\_filtered\Ridge\_model.pkl  
RMSE: 0.8314561612918128  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 5.375008802207079  
Sample 2: Real NR = 3.14, Predicted NR = 3.8658428251282797  
Sample 3: Real NR = 3.82, Predicted NR = 3.180267509916682  
Sample 4: Real NR = 3.84, Predicted NR = 3.372917385551846  
Sample 5: Real NR = 3.95, Predicted NR = 3.912580769420124  
  
  
Model: DecisionTreeRegressor  
Model saved as WS\_benefit\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 1.1169107581268032  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 4.070118764845606  
Sample 2: Real NR = 3.14, Predicted NR = 3.9491007804547  
Sample 3: Real NR = 3.82, Predicted NR = 3.565877163216831  
Sample 4: Real NR = 3.84, Predicted NR = 3.74  
Sample 5: Real NR = 3.95, Predicted NR = 3.95  
  
  
Model: GradientBoostingRegressor  
Model saved as WS\_benefit\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 0.7833358917734977  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.9860456075627977  
Sample 2: Real NR = 3.14, Predicted NR = 3.9860456075627977  
Sample 3: Real NR = 3.82, Predicted NR = 3.9860456075627977  
Sample 4: Real NR = 3.84, Predicted NR = 3.9738174416328134  
Sample 5: Real NR = 3.95, Predicted NR = 3.9860456075627977  
  
  
Model: RandomForestRegressor  
Model saved as WS\_benefit\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 0.9909656820775817  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.71  
Sample 2: Real NR = 3.14, Predicted NR = 4.02  
Sample 3: Real NR = 3.82, Predicted NR = 3.95  
Sample 4: Real NR = 3.84, Predicted NR = 3.72  
Sample 5: Real NR = 3.95, Predicted NR = 3.72  
  
  
Model: AdaBoostRegressor  
Model saved as WS\_benefit\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 0.8083848325456456  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.8030074652188675  
Sample 2: Real NR = 3.14, Predicted NR = 3.490932586811447  
Sample 3: Real NR = 3.82, Predicted NR = 3.8030074652188675  
Sample 4: Real NR = 3.84, Predicted NR = 3.8030074652188675  
Sample 5: Real NR = 3.95, Predicted NR = 3.8030074652188675  
  
  
Model: KNeighborsRegressor  
Model saved as WS\_benefit\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 0.8006476134637613  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 4.317772989480828  
Sample 2: Real NR = 3.14, Predicted NR = 3.822176111299627  
Sample 3: Real NR = 3.82, Predicted NR = 3.723520529351883  
Sample 4: Real NR = 3.84, Predicted NR = 3.9274418052256537  
Sample 5: Real NR = 3.95, Predicted NR = 3.832017645062776  
  
  
Model: MLPRegressor  
Model saved as WS\_benefit\_filtered\MLPRegressor\_model.pkl  
RMSE: 0.7788766410769133  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.989690927656253  
Sample 2: Real NR = 3.14, Predicted NR = 3.9900908110920144  
Sample 3: Real NR = 3.82, Predicted NR = 3.9901257486013786  
Sample 4: Real NR = 3.84, Predicted NR = 3.9898904071672354  
Sample 5: Real NR = 3.95, Predicted NR = 3.990417127144118  
  
  
Model: ElasticNet  
Model saved as WS\_benefit\_filtered\ElasticNet\_model.pkl  
RMSE: 0.7788100164488518  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.9900395882818684  
Sample 2: Real NR = 3.14, Predicted NR = 3.9900395882818684  
Sample 3: Real NR = 3.82, Predicted NR = 3.9900395882818684  
Sample 4: Real NR = 3.84, Predicted NR = 3.9900395882818684  
Sample 5: Real NR = 3.95, Predicted NR = 3.9900395882818684  
  
  
Model: SGDRegressor  
Model saved as WS\_benefit\_filtered\SGDRegressor\_model.pkl  
RMSE: 0.9680257162294521  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.8765503894232034  
Sample 2: Real NR = 3.14, Predicted NR = 3.677577435538908  
Sample 3: Real NR = 3.82, Predicted NR = 3.54626061829366  
Sample 4: Real NR = 3.84, Predicted NR = 3.567725995214532  
Sample 5: Real NR = 3.95, Predicted NR = 3.8816298088644943  
  
  
Model: SVR  
Model saved as WS\_benefit\_filtered\SVR\_model.pkl  
RMSE: 0.7834252422213258  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.96803359348535  
Sample 2: Real NR = 3.14, Predicted NR = 3.9680335934849165  
Sample 3: Real NR = 3.82, Predicted NR = 3.96803359348535  
Sample 4: Real NR = 3.84, Predicted NR = 3.96803359348535  
Sample 5: Real NR = 3.95, Predicted NR = 3.96803359348535  
  
  
Model: BayesianRidge  
Model saved as WS\_benefit\_filtered\BayesianRidge\_model.pkl  
RMSE: 0.7786601788108912  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.9909429250017463  
Sample 2: Real NR = 3.14, Predicted NR = 3.9898577880960038  
Sample 3: Real NR = 3.82, Predicted NR = 3.9897732458577817  
Sample 4: Real NR = 3.84, Predicted NR = 3.9900302228080404  
Sample 5: Real NR = 3.95, Predicted NR = 3.9897076732746894  
  
  
Model: KernelRidge  
Model saved as WS\_benefit\_filtered\KernelRidge\_model.pkl  
RMSE: 0.802547548252352  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 4.418080788379477  
Sample 2: Real NR = 3.14, Predicted NR = 3.7759497431898956  
Sample 3: Real NR = 3.82, Predicted NR = 3.5728668019998127  
Sample 4: Real NR = 3.84, Predicted NR = 3.740215930597695  
Sample 5: Real NR = 3.95, Predicted NR = 3.6813434997972427  
  
  
Model: LinearRegression  
Model saved as WS\_benefit\_filtered\LinearRegression\_model.pkl  
RMSE: 0.8311979419351034  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 4.127161514742986  
Sample 2: Real NR = 3.14, Predicted NR = 3.868534582744327  
Sample 3: Real NR = 3.82, Predicted NR = 3.760199482903918  
Sample 4: Real NR = 3.84, Predicted NR = 3.868534582744327  
Sample 5: Real NR = 3.95, Predicted NR = 4.101455047401207  
  
  
Model: RANSACRegressor  
Model saved as WS\_benefit\_filtered\RANSACRegressor\_model.pkl  
RMSE: 0.8789408476213671  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 3.7234112211243264  
Sample 2: Real NR = 3.14, Predicted NR = 3.8635048636955607  
Sample 3: Real NR = 3.82, Predicted NR = 3.882880929643351  
Sample 4: Real NR = 3.84, Predicted NR = 3.514374660774318  
Sample 5: Real NR = 3.95, Predicted NR = 4.17178667896578  
  
  
Model: TheilSenRegressor  
Model saved as WS\_benefit\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 0.9447622051720851  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = 5.7082204459339145  
Sample 2: Real NR = 3.14, Predicted NR = 3.900701655343604  
Sample 3: Real NR = 3.82, Predicted NR = 2.943415602128997  
Sample 4: Real NR = 3.84, Predicted NR = 3.1041603155040662  
Sample 5: Real NR = 3.95, Predicted NR = 3.976103161112392  
  
  
Model: TensorFlow  
Model saved as WS\_benefit\_filtered\TensorFlow\_model.h5  
RMSE: 1.4261565749474638  
Sample predictions:  
Sample 1: Real NR = 4.47, Predicted NR = [3.4946952]  
Sample 2: Real NR = 3.14, Predicted NR = [4.457201]  
Sample 3: Real NR = 3.82, Predicted NR = [3.950964]  
Sample 4: Real NR = 3.84, Predicted NR = [3.2694526]  
Sample 5: Real NR = 3.95, Predicted NR = [4.987297]

### Graph results





# Annex 3.3

## PR

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'saga'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'squared\_error', 'decisiontreeregressor\_\_max\_features': 1, 'decisiontreeregressor\_\_min\_samples\_split': 5, 'decisiontreeregressor\_\_splitter': 'random'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.01, 'gradientboostingregressor\_\_loss': 'huber', 'gradientboostingregressor\_\_n\_estimators': 250, 'gradientboostingregressor\_\_warm\_start': False}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'squared\_error', 'randomforestregressor\_\_max\_features': 1, 'randomforestregressor\_\_min\_samples\_split': 5, 'randomforestregressor\_\_n\_estimators': 50}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 0.1, 'adaboostregressor\_\_loss': 'square', 'adaboostregressor\_\_n\_estimators': 100}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'ball\_tree', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cityblock', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'uniform'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'identity', 'mlpregressor\_\_hidden\_layer\_sizes': (100, 100, 100), 'mlpregressor\_\_learning\_rate': 'adaptive', 'mlpregressor\_\_solver': 'adam'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': True, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.25, 'elasticnet\_\_positive': False, 'elasticnet\_\_precompute': False, 'elasticnet\_\_selection': 'random', 'elasticnet\_\_warm\_start': True}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'invscaling', 'sgdregressor\_\_loss': 'squared\_error', 'sgdregressor\_\_penalty': 'l2', 'sgdregressor\_\_warm\_start': False}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 'scale', 'svr\_\_kernel': 'poly', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-07, 'bayesianridge\_\_alpha\_2': 1e-05, 'bayesianridge\_\_lambda\_1': 1e-05, 'bayesianridge\_\_lambda\_2': 1e-07}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': False}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'squared\_error', 'ransacregressor\_\_max\_trials': 10, 'ransacregressor\_\_min\_samples': 2}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 10, 'theilsenregressor\_\_n\_subsamples': 25}

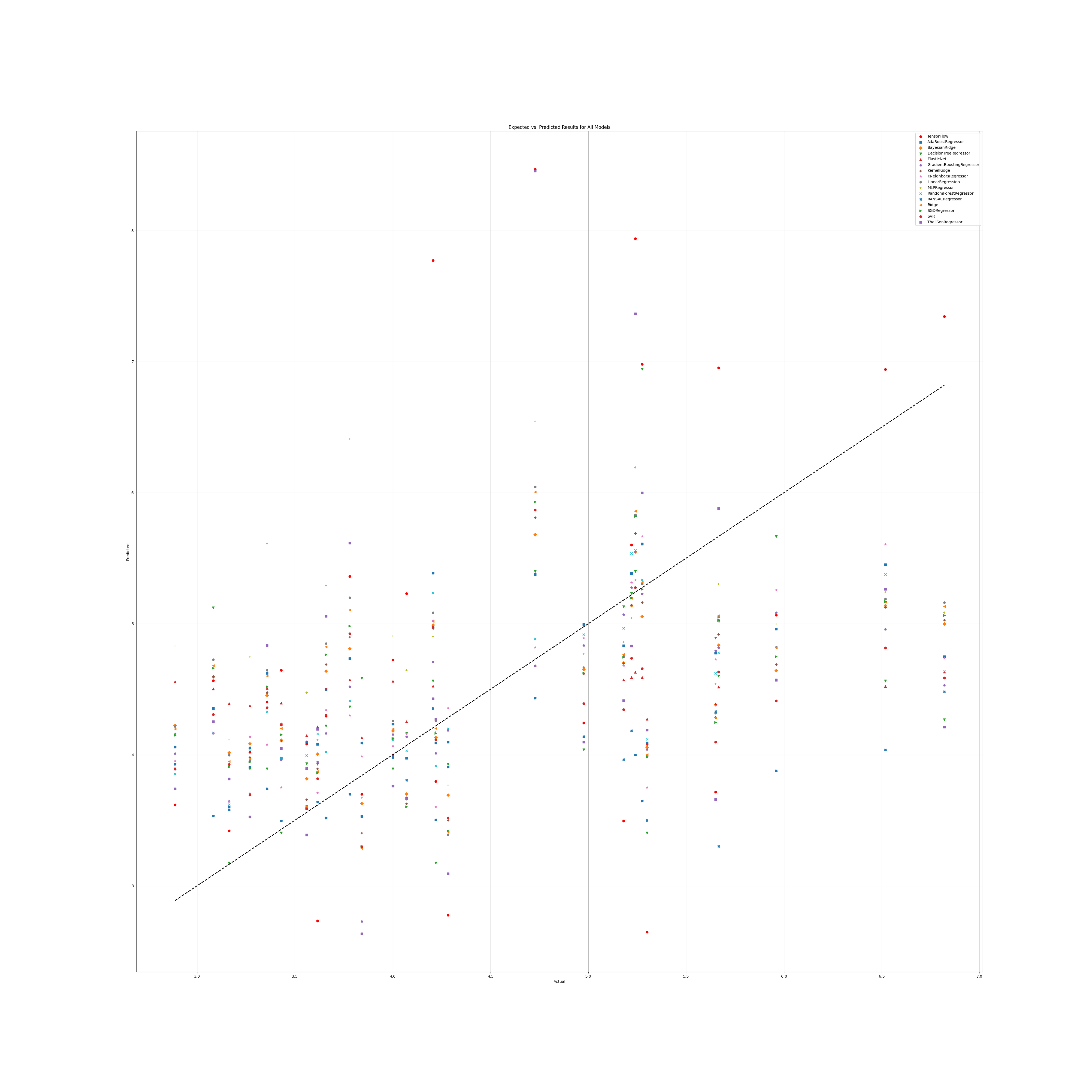
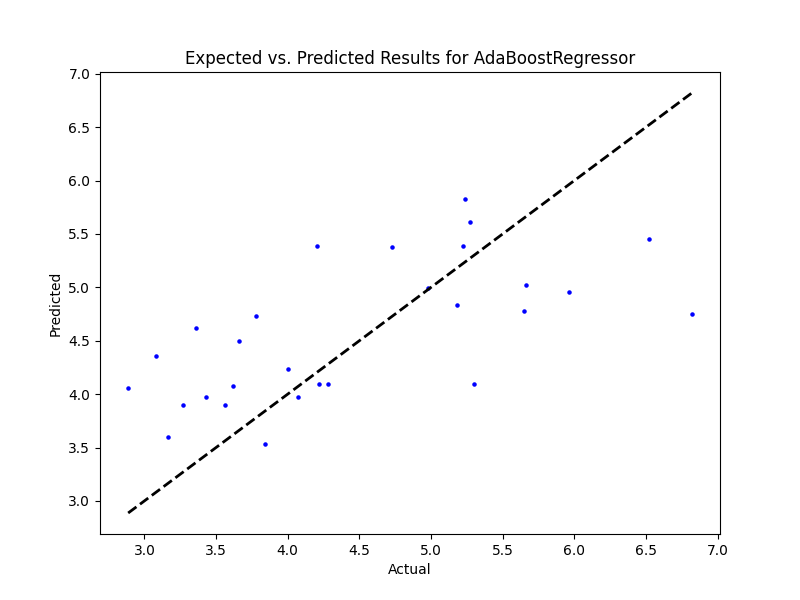
**TensorFlow ()**

TensorFlow RMSE: 1.5623859725963753

### Prediction results

Model: Ridge  
Model saved as PR\_filtered\Ridge\_model.pkl  
RMSE: 0.9414530442623293  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.6668600921376076  
Sample 2: Real NR = 3.56, Predicted NR = 3.6035542404542302  
Sample 3: Real NR = 4.0, Predicted NR = 4.196111712873428  
Sample 4: Real NR = 6.82, Predicted NR = 5.13321359114227  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.9490534139602285  
  
  
Model: DecisionTreeRegressor  
Model saved as PR\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 1.009503396386322  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 4.166243597090477  
Sample 2: Real NR = 3.56, Predicted NR = 3.9329973240571783  
Sample 3: Real NR = 4.0, Predicted NR = 3.8929849036275197  
Sample 4: Real NR = 6.82, Predicted NR = 4.2675  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.173429669509053  
  
  
Model: GradientBoostingRegressor  
Model saved as PR\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 0.8145982320701887  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 4.137111976522874  
Sample 2: Real NR = 3.56, Predicted NR = 3.8953067804574126  
Sample 3: Real NR = 4.0, Predicted NR = 4.155719659814404  
Sample 4: Real NR = 6.82, Predicted NR = 4.5312019887656145  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.6441960979541146  
  
  
Model: RandomForestRegressor  
Model saved as PR\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 0.7740911147200868  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 4.0296875620610555  
Sample 2: Real NR = 3.56, Predicted NR = 3.9943549099650335  
Sample 3: Real NR = 4.0, Predicted NR = 4.110692169439252  
Sample 4: Real NR = 6.82, Predicted NR = 4.635067989901376  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.6202420075653885  
  
  
Model: AdaBoostRegressor  
Model saved as PR\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 0.8264593777603052  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.9750926930808057  
Sample 2: Real NR = 3.56, Predicted NR = 3.895192569515227  
Sample 3: Real NR = 4.0, Predicted NR = 4.234802840074878  
Sample 4: Real NR = 6.82, Predicted NR = 4.74904941243889  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.6002092950203064  
  
  
Model: KNeighborsRegressor  
Model saved as PR\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 0.7516526589867194  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 4.159396636053344  
Sample 2: Real NR = 3.56, Predicted NR = 4.101805485599004  
Sample 3: Real NR = 4.0, Predicted NR = 4.067478933443254  
Sample 4: Real NR = 6.82, Predicted NR = 4.737270927727581  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.6443721923126553  
  
  
Model: MLPRegressor  
Model saved as PR\_filtered\MLPRegressor\_model.pkl  
RMSE: 1.198474905945482  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 4.6447838020035155  
Sample 2: Real NR = 3.56, Predicted NR = 4.47410333860162  
Sample 3: Real NR = 4.0, Predicted NR = 4.904564611130194  
Sample 4: Real NR = 6.82, Predicted NR = 5.084514018665849  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 4.113405695163608  
  
  
Model: ElasticNet  
Model saved as PR\_filtered\ElasticNet\_model.pkl  
RMSE: 1.0136264823739376  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 4.253251895748107  
Sample 2: Real NR = 3.56, Predicted NR = 4.1469728684760065  
Sample 3: Real NR = 4.0, Predicted NR = 4.561338462393676  
Sample 4: Real NR = 6.82, Predicted NR = 4.632757105898122  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 4.390396196235467  
  
  
Model: SGDRegressor  
Model saved as PR\_filtered\SGDRegressor\_model.pkl  
RMSE: 0.9287023497454953  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.6039904246650236  
Sample 2: Real NR = 3.56, Predicted NR = 3.5911731652885903  
Sample 3: Real NR = 4.0, Predicted NR = 4.124900471258661  
Sample 4: Real NR = 6.82, Predicted NR = 5.062726601888814  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.907090637900649  
  
  
Model: SVR  
Model saved as PR\_filtered\SVR\_model.pkl  
RMSE: 0.9732253472005621  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.6724053648993875  
Sample 2: Real NR = 3.56, Predicted NR = 3.5880798043997464  
Sample 3: Real NR = 4.0, Predicted NR = 3.999921727395699  
Sample 4: Real NR = 6.82, Predicted NR = 4.585887008082266  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.925590108533047  
  
  
Model: BayesianRidge  
Model saved as PR\_filtered\BayesianRidge\_model.pkl  
RMSE: 0.9011312996657181  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.701930704754554  
Sample 2: Real NR = 3.56, Predicted NR = 3.8174958951221574  
Sample 3: Real NR = 4.0, Predicted NR = 4.182611170737088  
Sample 4: Real NR = 6.82, Predicted NR = 4.9997000050823  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 4.015419226983606  
  
  
Model: KernelRidge  
Model saved as PR\_filtered\KernelRidge\_model.pkl  
RMSE: 0.9137577106111415  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.6259196320929736  
Sample 2: Real NR = 3.56, Predicted NR = 3.6582036867552294  
Sample 3: Real NR = 4.0, Predicted NR = 4.128237162079709  
Sample 4: Real NR = 6.82, Predicted NR = 5.027987308264808  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.9277472205528667  
  
  
Model: LinearRegression  
Model saved as PR\_filtered\LinearRegression\_model.pkl  
RMSE: 0.9557838843183363  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.661650980325474  
Sample 2: Real NR = 3.56, Predicted NR = 3.6100303093564334  
Sample 3: Real NR = 4.0, Predicted NR = 4.259918976198943  
Sample 4: Real NR = 6.82, Predicted NR = 5.161024109463913  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.9966929423020874  
  
  
Model: RANSACRegressor  
Model saved as PR\_filtered\RANSACRegressor\_model.pkl  
RMSE: 1.1557033497378388  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.8045266175019283  
Sample 2: Real NR = 3.56, Predicted NR = 4.099202764717206  
Sample 3: Real NR = 4.0, Predicted NR = 3.9799984686592618  
Sample 4: Real NR = 6.82, Predicted NR = 4.4827013998765  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.581260356231873  
  
  
Model: TheilSenRegressor  
Model saved as PR\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 1.3363559818160895  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = 3.666570629048902  
Sample 2: Real NR = 3.56, Predicted NR = 3.3895412181892652  
Sample 3: Real NR = 4.0, Predicted NR = 3.761748395928905  
Sample 4: Real NR = 6.82, Predicted NR = 4.212300799656813  
Sample 5: Real NR = 3.163892700892594, Predicted NR = 3.816547632745216  
  
  
Model: TensorFlow  
Model saved as PR\_filtered\TensorFlow\_model.h5  
RMSE: 1.5623859725963753  
Sample predictions:  
Sample 1: Real NR = 4.07, Predicted NR = [5.2295523]  
Sample 2: Real NR = 3.56, Predicted NR = [4.082267]  
Sample 3: Real NR = 4.0, Predicted NR = [4.724395]  
Sample 4: Real NR = 6.82, Predicted NR = [7.345411]  
Sample 5: Real NR = 3.163892700892594, Predicted NR = [3.4205678]

### Graph results



## PR Benefit

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'sparse\_cg'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'friedman\_mse', 'decisiontreeregressor\_\_max\_features': 1, 'decisiontreeregressor\_\_min\_samples\_split': 4, 'decisiontreeregressor\_\_splitter': 'random'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.1, 'gradientboostingregressor\_\_loss': 'absolute\_error', 'gradientboostingregressor\_\_n\_estimators': 250, 'gradientboostingregressor\_\_warm\_start': True}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'squared\_error', 'randomforestregressor\_\_max\_features': 'sqrt', 'randomforestregressor\_\_min\_samples\_split': 5, 'randomforestregressor\_\_n\_estimators': 1}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 0.0001, 'adaboostregressor\_\_loss': 'square', 'adaboostregressor\_\_n\_estimators': 50}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'brute', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cosine', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'uniform'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'tanh', 'mlpregressor\_\_hidden\_layer\_sizes': (50, 50, 50), 'mlpregressor\_\_learning\_rate': 'adaptive', 'mlpregressor\_\_solver': 'sgd'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': True, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.25, 'elasticnet\_\_positive': False, 'elasticnet\_\_precompute': True, 'elasticnet\_\_selection': 'random', 'elasticnet\_\_warm\_start': False}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'invscaling', 'sgdregressor\_\_loss': 'squared\_error', 'sgdregressor\_\_penalty': 'l1', 'sgdregressor\_\_warm\_start': True}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 'scale', 'svr\_\_kernel': 'poly', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-07, 'bayesianridge\_\_alpha\_2': 1e-05, 'bayesianridge\_\_lambda\_1': 1e-05, 'bayesianridge\_\_lambda\_2': 1e-05}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': True}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'absolute\_error', 'ransacregressor\_\_max\_trials': 50, 'ransacregressor\_\_min\_samples': 1}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 100, 'theilsenregressor\_\_n\_subsamples': 25}

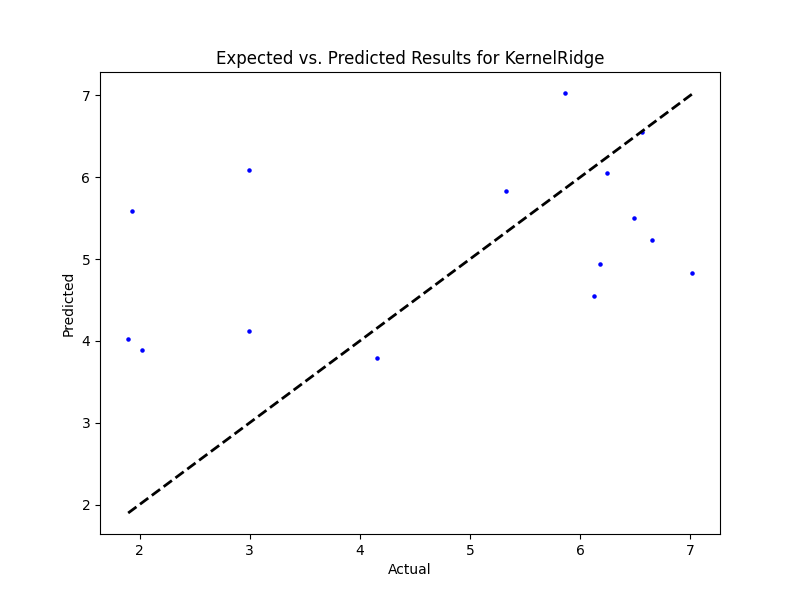
**TensorFlow()**

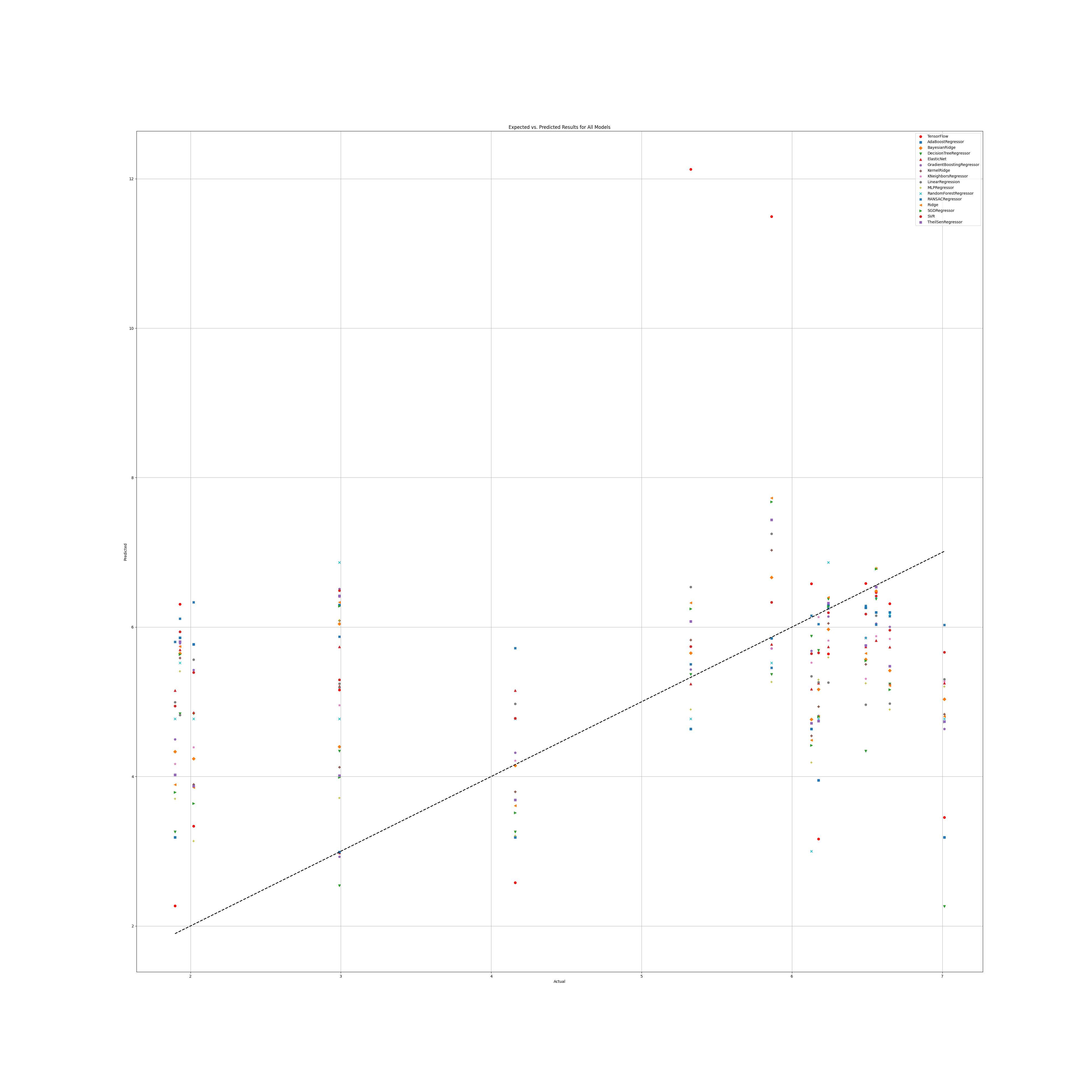
TensorFlow RMSE: 2.928123193898337

### Prediction results

Model: Ridge  
Model saved as PR\_benefit\_filtered\Ridge\_model.pkl  
RMSE: 1.8456546442618005  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 3.855688250645724  
Sample 2: Real NR = 6.56, Predicted NR = 6.784718284984148  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.612057483559189  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 4.805688303166861  
Sample 5: Real NR = 6.65, Predicted NR = 5.216426294289889  
  
  
Model: DecisionTreeRegressor  
Model saved as PR\_benefit\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 1.8427531420151655  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 4.843333333333333  
Sample 2: Real NR = 6.56, Predicted NR = 6.377599561352882  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.2595636615491173  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 2.262433007371251  
Sample 5: Real NR = 6.65, Predicted NR = 5.24  
  
  
Model: GradientBoostingRegressor  
Model saved as PR\_benefit\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 1.9056994571046753  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 5.425220959691096  
Sample 2: Real NR = 6.56, Predicted NR = 6.045378916244852  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 4.32064074624981  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 4.63553091627332  
Sample 5: Real NR = 6.65, Predicted NR = 6.003254250808956  
  
  
Model: RandomForestRegressor  
Model saved as PR\_benefit\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 2.090390333146172  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 4.770788179614453  
Sample 2: Real NR = 6.56, Predicted NR = 6.402677680483868  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 4.770788179614453  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 4.770788179614453  
Sample 5: Real NR = 6.65, Predicted NR = 6.169679383585363  
  
  
Model: AdaBoostRegressor  
Model saved as PR\_benefit\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 2.09367938174643  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 5.77  
Sample 2: Real NR = 6.56, Predicted NR = 6.196423018012471  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.185578513986687  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 3.185578513986687  
Sample 5: Real NR = 6.65, Predicted NR = 6.196423018012471  
  
  
Model: KNeighborsRegressor  
Model saved as PR\_benefit\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 1.769300232920453  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 4.393639666232417  
Sample 2: Real NR = 6.56, Predicted NR = 5.877391525535529  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 4.21217023136162  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 5.277261085734074  
Sample 5: Real NR = 6.65, Predicted NR = 5.843830857010934  
  
  
Model: MLPRegressor  
Model saved as PR\_benefit\_filtered\MLPRegressor\_model.pkl  
RMSE: 1.6556012148736177  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 3.1362736643451163  
Sample 2: Real NR = 6.56, Predicted NR = 6.791304374334875  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.2100306299100994  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 5.204539502280477  
Sample 5: Real NR = 6.65, Predicted NR = 4.896613717882464  
  
  
Model: ElasticNet  
Model saved as PR\_benefit\_filtered\ElasticNet\_model.pkl  
RMSE: 1.8862641161009368  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 4.8570376708826695  
Sample 2: Real NR = 6.56, Predicted NR = 5.822239918731109  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 5.150331063081614  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 5.250211903309277  
Sample 5: Real NR = 6.65, Predicted NR = 5.732043694721569  
  
  
Model: SGDRegressor  
Model saved as PR\_benefit\_filtered\SGDRegressor\_model.pkl  
RMSE: 1.8160486059185204  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 3.6399069259559305  
Sample 2: Real NR = 6.56, Predicted NR = 6.776690680649487  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.51460435835421  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 4.731683976511796  
Sample 5: Real NR = 6.65, Predicted NR = 5.163225979694507  
  
  
Model: SVR  
Model saved as PR\_benefit\_filtered\SVR\_model.pkl  
RMSE: 1.9660088756473901  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 5.3928673854409785  
Sample 2: Real NR = 6.56, Predicted NR = 6.417543667921111  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 4.779504441289926  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 5.664212267072398  
Sample 5: Real NR = 6.65, Predicted NR = 5.960016515699077  
  
  
Model: BayesianRidge  
Model saved as PR\_benefit\_filtered\BayesianRidge\_model.pkl  
RMSE: 1.751194086691448  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 4.240048189706345  
Sample 2: Real NR = 6.56, Predicted NR = 6.481071886905376  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 4.147581981824204  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 5.036984460339915  
Sample 5: Real NR = 6.65, Predicted NR = 5.418010072414029  
  
  
Model: KernelRidge  
Model saved as PR\_benefit\_filtered\KernelRidge\_model.pkl  
RMSE: 1.7479282161616718  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 3.8945430256343396  
Sample 2: Real NR = 6.56, Predicted NR = 6.54783567214176  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.7961871431746275  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 4.833127954813017  
Sample 5: Real NR = 6.65, Predicted NR = 5.237758391720566  
  
  
Model: LinearRegression  
Model saved as PR\_benefit\_filtered\LinearRegression\_model.pkl  
RMSE: 1.9175624498524109  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 5.563997289137757  
Sample 2: Real NR = 6.56, Predicted NR = 6.153363749060477  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 4.974101574469261  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 5.301193122670451  
Sample 5: Real NR = 6.65, Predicted NR = 4.978510979441057  
  
  
Model: RANSACRegressor  
Model saved as PR\_benefit\_filtered\RANSACRegressor\_model.pkl  
RMSE: 2.2447260367372586  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 6.333062263299267  
Sample 2: Real NR = 6.56, Predicted NR = 6.034135900650113  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 5.717650292290312  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 6.02979149171645  
Sample 5: Real NR = 6.65, Predicted NR = 6.146150503251641  
  
  
Model: TheilSenRegressor  
Model saved as PR\_benefit\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 1.8294291904584887  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = 3.8742823352069813  
Sample 2: Real NR = 6.56, Predicted NR = 6.537869003310232  
Sample 3: Real NR = 4.159067297702875, Predicted NR = 3.6885095827442527  
Sample 4: Real NR = 7.013614551211226, Predicted NR = 4.733882427292508  
Sample 5: Real NR = 6.65, Predicted NR = 5.476995526386787  
  
  
Model: TensorFlow  
Model saved as PR\_benefit\_filtered\TensorFlow\_model.h5  
RMSE: 2.928123193898337  
Sample predictions:  
Sample 1: Real NR = 2.02, Predicted NR = [3.3361132]  
Sample 2: Real NR = 6.56, Predicted NR = [6.4629197]  
Sample 3: Real NR = 4.159067297702875, Predicted NR = [2.580814]  
Sample 4: Real NR = 7.013614551211226, Predicted NR = [3.4530346]  
Sample 5: Real NR = 6.65, Predicted NR = [6.3149943]

### Graph results





# Annex 3.4

## SR

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'cholesky'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'squared\_error', 'decisiontreeregressor\_\_max\_features': 3, 'decisiontreeregressor\_\_min\_samples\_split': 4, 'decisiontreeregressor\_\_splitter': 'random'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.1, 'gradientboostingregressor\_\_loss': 'absolute\_error', 'gradientboostingregressor\_\_n\_estimators': 25, 'gradientboostingregressor\_\_warm\_start': True}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'squared\_error', 'randomforestregressor\_\_max\_features': 3, 'randomforestregressor\_\_min\_samples\_split': 5, 'randomforestregressor\_\_n\_estimators': 100}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 1.0, 'adaboostregressor\_\_loss': 'exponential', 'adaboostregressor\_\_n\_estimators': 20}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'ball\_tree', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cityblock', 'kneighborsregressor\_\_n\_neighbors': 10, 'kneighborsregressor\_\_weights': 'uniform'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'relu', 'mlpregressor\_\_hidden\_layer\_sizes': (100, 100, 100, 100), 'mlpregressor\_\_learning\_rate': 'adaptive', 'mlpregressor\_\_solver': 'sgd'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': False, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.25, 'elasticnet\_\_positive': False, 'elasticnet\_\_precompute': True, 'elasticnet\_\_selection': 'random', 'elasticnet\_\_warm\_start': False}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'constant', 'sgdregressor\_\_loss': 'squared\_epsilon\_insensitive', 'sgdregressor\_\_penalty': 'elasticnet', 'sgdregressor\_\_warm\_start': True}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 'auto', 'svr\_\_kernel': 'rbf', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-05, 'bayesianridge\_\_alpha\_2': 1e-07, 'bayesianridge\_\_lambda\_1': 1e-07, 'bayesianridge\_\_lambda\_2': 1e-05}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': False}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'squared\_error', 'ransacregressor\_\_max\_trials': 10, 'ransacregressor\_\_min\_samples': 1}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 1000, 'theilsenregressor\_\_n\_subsamples': 25}

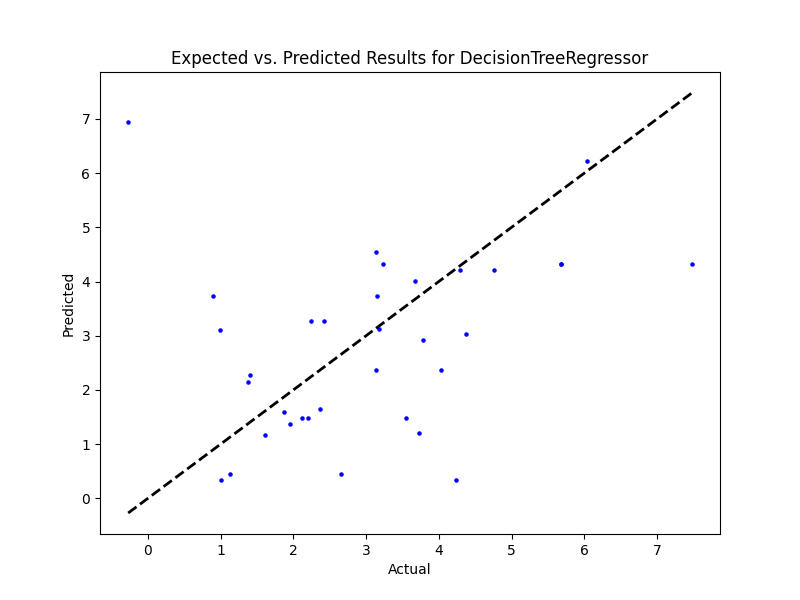
**TensorFlow ()**

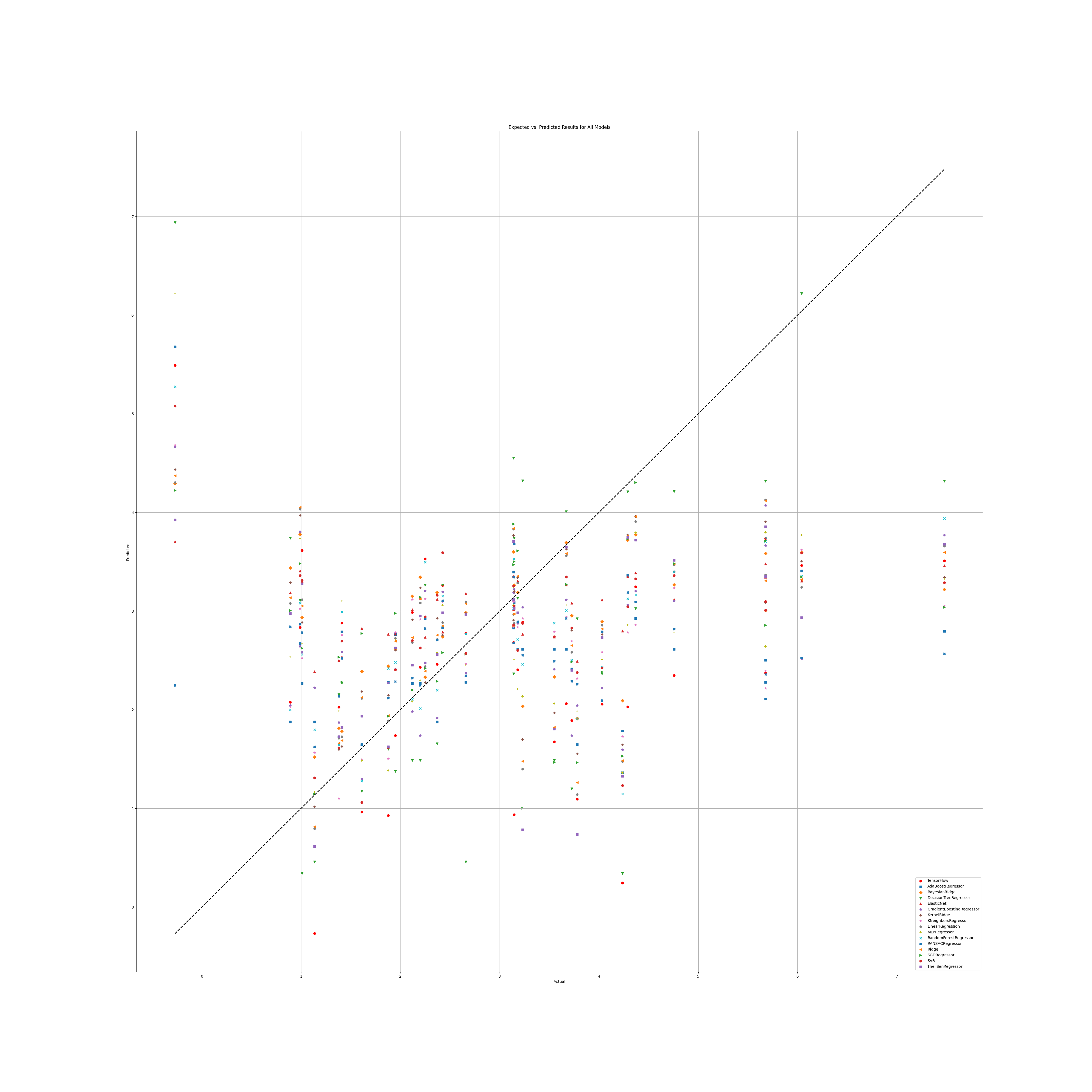
TensorFlow RMSE: 2.0687468799830544

### Prediction results

Model: Ridge  
Model saved as SR\_filtered\Ridge\_model.pkl  
RMSE: 1.704395040129048  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 4.120478534053572  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 2.9731177362945527  
Sample 3: Real NR = 1.38, Predicted NR = 1.6615860955022796  
Sample 4: Real NR = 1.61, Predicted NR = 2.1273894945791048  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.9586549505366233  
  
  
Model: DecisionTreeRegressor  
Model saved as SR\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 1.9161895912274403  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 4.3177270908363345  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.74  
Sample 3: Real NR = 1.38, Predicted NR = 2.155  
Sample 4: Real NR = 1.61, Predicted NR = 1.175  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.026876075731497  
  
  
Model: GradientBoostingRegressor  
Model saved as SR\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 1.659076656980171  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 4.071410952926566  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.2237820019377708  
Sample 3: Real NR = 1.38, Predicted NR = 1.8718829243543955  
Sample 4: Real NR = 1.61, Predicted NR = 1.2996634463363754  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.203298122719133  
  
  
Model: RandomForestRegressor  
Model saved as SR\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 1.676956206203133  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.743743691393371  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.5294885766167066  
Sample 3: Real NR = 1.38, Predicted NR = 1.644283678336366  
Sample 4: Real NR = 1.61, Predicted NR = 1.278115216491748  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.164666037353949  
  
  
Model: AdaBoostRegressor  
Model saved as SR\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 1.8956858854224272  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 2.503751706792872  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.0880430766684186  
Sample 3: Real NR = 1.38, Predicted NR = 1.721855140665279  
Sample 4: Real NR = 1.61, Predicted NR = 1.6476314599993758  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 2.9258415834492144  
  
  
Model: KNeighborsRegressor  
Model saved as SR\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 1.7671155987851959  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 2.2175291588235977  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.03468381707244  
Sample 3: Real NR = 1.38, Predicted NR = 1.101420418769918  
Sample 4: Real NR = 1.61, Predicted NR = 1.4976398414787604  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 2.8594461457560647  
  
  
Model: MLPRegressor  
Model saved as SR\_filtered\MLPRegressor\_model.pkl  
RMSE: 1.8935906477083286  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.799276484855395  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 2.509975451250035  
Sample 3: Real NR = 1.38, Predicted NR = 1.9887927358466473  
Sample 4: Real NR = 1.61, Predicted NR = 1.4853834013873317  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.794149974056236  
  
  
Model: ElasticNet  
Model saved as SR\_filtered\ElasticNet\_model.pkl  
RMSE: 1.5966651808542822  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.479834498887749  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 2.862665399870068  
Sample 3: Real NR = 1.38, Predicted NR = 2.499505619378715  
Sample 4: Real NR = 1.61, Predicted NR = 2.8238504638911093  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.3899905445040766  
  
  
Model: SGDRegressor  
Model saved as SR\_filtered\SGDRegressor\_model.pkl  
RMSE: 1.7744164417558232  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.712106027229107  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.5033229867855806  
Sample 3: Real NR = 1.38, Predicted NR = 2.5334833712156035  
Sample 4: Real NR = 1.61, Predicted NR = 2.7747950845003464  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 4.304446004484192  
  
  
Model: SVR  
Model saved as SR\_filtered\SVR\_model.pkl  
RMSE: 1.7726982270719094  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.0966732037718843  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.0525492226559265  
Sample 3: Real NR = 1.38, Predicted NR = 1.6135040901331577  
Sample 4: Real NR = 1.61, Predicted NR = 1.062050631100679  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.3274013663333912  
  
  
Model: BayesianRidge  
Model saved as SR\_filtered\BayesianRidge\_model.pkl  
RMSE: 1.67698609898832  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.585327729325311  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.263863610800425  
Sample 3: Real NR = 1.38, Predicted NR = 1.8146209827128053  
Sample 4: Real NR = 1.61, Predicted NR = 2.390252667357988  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.776959517271657  
  
  
Model: KernelRidge  
Model saved as SR\_filtered\KernelRidge\_model.pkl  
RMSE: 1.7024668397982028  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.9074039161274587  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.201182747332939  
Sample 3: Real NR = 1.38, Predicted NR = 1.5958733268120855  
Sample 4: Real NR = 1.61, Predicted NR = 2.185957761941492  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.962745202724633  
  
  
Model: LinearRegression  
Model saved as SR\_filtered\LinearRegression\_model.pkl  
RMSE: 1.7048324907355388  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 4.12661944365046  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 2.876674250201104  
Sample 3: Real NR = 1.38, Predicted NR = 1.711409022081715  
Sample 4: Real NR = 1.61, Predicted NR = 2.11495476869404  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.9086167490871935  
  
  
Model: RANSACRegressor  
Model saved as SR\_filtered\RANSACRegressor\_model.pkl  
RMSE: 1.7470948996912832  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 2.1112912422926344  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.683141342068237  
Sample 3: Real NR = 1.38, Predicted NR = 2.1369551600547823  
Sample 4: Real NR = 1.61, Predicted NR = 1.645881634984312  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.0911498316098607  
  
  
Model: TheilSenRegressor  
Model saved as SR\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 1.7407533068910521  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = 3.8577114306213964  
Sample 2: Real NR = 3.146157559409307, Predicted NR = 3.0291865640029245  
Sample 3: Real NR = 1.38, Predicted NR = 1.7286192358929022  
Sample 4: Real NR = 1.61, Predicted NR = 1.9361155088798065  
Sample 5: Real NR = 4.369850565010858, Predicted NR = 3.7212355905636842  
  
  
Model: TensorFlow  
Model saved as SR\_filtered\TensorFlow\_model.h5  
RMSE: 2.0687468799830544  
Sample predictions:  
Sample 1: Real NR = 5.678271308523408, Predicted NR = [2.3731039]  
Sample 2: Real NR = 3.146157559409307, Predicted NR = [0.93680096]  
Sample 3: Real NR = 1.38, Predicted NR = [2.0281885]  
Sample 4: Real NR = 1.61, Predicted NR = [0.9650718]  
Sample 5: Real NR = 4.369850565010858, Predicted NR = [3.2473586]

### Graph results





## SR Benefit

### Grid search results

**Ridge()**  
Best hyperparameters for Ridge: {'ridge\_\_alpha': 1.0, 'ridge\_\_solver': 'sparse\_cg'}

**DecisionTreeRegressor()**  
Best hyperparameters for DecisionTreeRegressor: {'decisiontreeregressor\_\_criterion': 'absolute\_error', 'decisiontreeregressor\_\_max\_features': 2, 'decisiontreeregressor\_\_min\_samples\_split': 5, 'decisiontreeregressor\_\_splitter': 'best'}

**GradientBoostingRegressor()**  
Best hyperparameters for GradientBoostingRegressor: {'gradientboostingregressor\_\_learning\_rate': 0.1, 'gradientboostingregressor\_\_loss': 'huber', 'gradientboostingregressor\_\_n\_estimators': 100, 'gradientboostingregressor\_\_warm\_start': False}

**RandomForestRegressor()**  
Best hyperparameters for RandomForestRegressor: {'randomforestregressor\_\_criterion': 'absolute\_error', 'randomforestregressor\_\_max\_features': 'log2', 'randomforestregressor\_\_min\_samples\_split': 2, 'randomforestregressor\_\_n\_estimators': 50}

**AdaBoostRegressor()**  
Best hyperparameters for AdaBoostRegressor: {'adaboostregressor\_\_learning\_rate': 1.0, 'adaboostregressor\_\_loss': 'square', 'adaboostregressor\_\_n\_estimators': 20}

**KNeighborsRegressor()**  
Best hyperparameters for KNeighborsRegressor: {'kneighborsregressor\_\_algorithm': 'brute', 'kneighborsregressor\_\_leaf\_size': 5, 'kneighborsregressor\_\_metric': 'cosine', 'kneighborsregressor\_\_n\_neighbors': 5, 'kneighborsregressor\_\_weights': 'distance'}

**MLPRegressor()**  
Best hyperparameters for MLPRegressor: {'mlpregressor\_\_activation': 'identity', 'mlpregressor\_\_hidden\_layer\_sizes': (50, 50, 50), 'mlpregressor\_\_learning\_rate': 'adaptive', 'mlpregressor\_\_solver': 'sgd'}

**ElasticNet()**  
Best hyperparameters for ElasticNet: {'elasticnet\_\_copy\_X': True, 'elasticnet\_\_fit\_intercept': True, 'elasticnet\_\_l1\_ratio': 0.25, 'elasticnet\_\_positive': True, 'elasticnet\_\_precompute': True, 'elasticnet\_\_selection': 'cyclic', 'elasticnet\_\_warm\_start': True}

**SGDRegressor()**  
Best hyperparameters for SGDRegressor: {'sgdregressor\_\_learning\_rate': 'invscaling', 'sgdregressor\_\_loss': 'squared\_epsilon\_insensitive', 'sgdregressor\_\_penalty': 'l2', 'sgdregressor\_\_warm\_start': True}

**SVR(cache\_size=1000)**  
Best hyperparameters for SVR: {'svr\_\_degree': 1, 'svr\_\_gamma': 'scale', 'svr\_\_kernel': 'rbf', 'svr\_\_shrinking': True}

**BayesianRidge(max\_iter=1000)**  
Best hyperparameters for BayesianRidge: {'bayesianridge\_\_alpha\_1': 1e-05, 'bayesianridge\_\_alpha\_2': 1e-07, 'bayesianridge\_\_lambda\_1': 1e-07, 'bayesianridge\_\_lambda\_2': 1e-05}

**KernelRidge()**  
Best hyperparameters for KernelRidge: {'kernelridge\_\_alpha': 1.0, 'kernelridge\_\_coef0': 1.0, 'kernelridge\_\_degree': 1, 'kernelridge\_\_kernel': 'poly'}

**LinearRegression()**  
Best hyperparameters for LinearRegression: {'linearregression\_\_copy\_X': True, 'linearregression\_\_fit\_intercept': True, 'linearregression\_\_positive': False}

**RANSACRegressor()**  
Best hyperparameters for RANSACRegressor: {'ransacregressor\_\_loss': 'squared\_error', 'ransacregressor\_\_max\_trials': 1, 'ransacregressor\_\_min\_samples': 5}

**TheilSenRegressor()**  
Best hyperparameters for TheilSenRegressor: {'theilsenregressor\_\_max\_subpopulation': 10000, 'theilsenregressor\_\_n\_subsamples': 25}

**TensorFlow()**

TensorFlow RMSE: 0.7431941166450133

### Prediction results

Model: Ridge  
Model saved as SR\_benefit\_filtered\Ridge\_model.pkl  
RMSE: 0.5040009512558744  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.769386540395349  
Sample 2: Real NR = 4.24, Predicted NR = 4.870260952836919  
Sample 3: Real NR = 4.15, Predicted NR = 4.94848920301992  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.555025837148188  
Sample 5: Real NR = 4.17, Predicted NR = 4.488144386610655  
  
  
Model: DecisionTreeRegressor  
Model saved as SR\_benefit\_filtered\DecisionTreeRegressor\_model.pkl  
RMSE: 0.5130222738799957  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.51  
Sample 2: Real NR = 4.24, Predicted NR = 4.662535614486191  
Sample 3: Real NR = 4.15, Predicted NR = 5.17  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 5.432702439402354  
Sample 5: Real NR = 4.17, Predicted NR = 4.42  
  
  
Model: GradientBoostingRegressor  
Model saved as SR\_benefit\_filtered\GradientBoostingRegressor\_model.pkl  
RMSE: 0.41893392116300543  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.5622050464727275  
Sample 2: Real NR = 4.24, Predicted NR = 4.548848528425731  
Sample 3: Real NR = 4.15, Predicted NR = 4.56993474549033  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.358563594876846  
Sample 5: Real NR = 4.17, Predicted NR = 4.554705966707614  
  
  
Model: RandomForestRegressor  
Model saved as SR\_benefit\_filtered\RandomForestRegressor\_model.pkl  
RMSE: 0.3496530348488033  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.639531429385379  
Sample 2: Real NR = 4.24, Predicted NR = 4.682852473985689  
Sample 3: Real NR = 4.15, Predicted NR = 4.537066244831288  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.722843065315574  
Sample 5: Real NR = 4.17, Predicted NR = 4.627178816598797  
  
  
Model: AdaBoostRegressor  
Model saved as SR\_benefit\_filtered\AdaBoostRegressor\_model.pkl  
RMSE: 0.4281895901622235  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.788917499013216  
Sample 2: Real NR = 4.24, Predicted NR = 4.612581310350363  
Sample 3: Real NR = 4.15, Predicted NR = 4.252351752530764  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.46238164912086  
Sample 5: Real NR = 4.17, Predicted NR = 4.426491402426744  
  
  
Model: KNeighborsRegressor  
Model saved as SR\_benefit\_filtered\KNeighborsRegressor\_model.pkl  
RMSE: 0.409439840209338  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.831810186932994  
Sample 2: Real NR = 4.24, Predicted NR = 4.578366480236642  
Sample 3: Real NR = 4.15, Predicted NR = 4.7381598652296955  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.5110933858076505  
Sample 5: Real NR = 4.17, Predicted NR = 4.708875794835  
  
  
Model: MLPRegressor  
Model saved as SR\_benefit\_filtered\MLPRegressor\_model.pkl  
RMSE: 0.5195667885431746  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.787026605140353  
Sample 2: Real NR = 4.24, Predicted NR = 4.880115079549912  
Sample 3: Real NR = 4.15, Predicted NR = 4.944925062688541  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.565953002124368  
Sample 5: Real NR = 4.17, Predicted NR = 4.5035966129382885  
  
  
Model: ElasticNet  
Model saved as SR\_benefit\_filtered\ElasticNet\_model.pkl  
RMSE: 0.32788511512878094  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.619455688749345  
Sample 2: Real NR = 4.24, Predicted NR = 4.619455688749345  
Sample 3: Real NR = 4.15, Predicted NR = 4.619455688749345  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.619455688749345  
Sample 5: Real NR = 4.17, Predicted NR = 4.619455688749345  
  
  
Model: SGDRegressor  
Model saved as SR\_benefit\_filtered\SGDRegressor\_model.pkl  
RMSE: 0.47838915747037286  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.7333133514798025  
Sample 2: Real NR = 4.24, Predicted NR = 4.816686455858914  
Sample 3: Real NR = 4.15, Predicted NR = 4.822766077793564  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.478963222565211  
Sample 5: Real NR = 4.17, Predicted NR = 4.510275797076521  
  
  
Model: SVR  
Model saved as SR\_benefit\_filtered\SVR\_model.pkl  
RMSE: 0.42126429770231805  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.5258965882323325  
Sample 2: Real NR = 4.24, Predicted NR = 4.84543258151524  
Sample 3: Real NR = 4.15, Predicted NR = 4.780595999506473  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.589542924929  
Sample 5: Real NR = 4.17, Predicted NR = 4.485705965390235  
  
  
Model: BayesianRidge  
Model saved as SR\_benefit\_filtered\BayesianRidge\_model.pkl  
RMSE: 0.43628052876385  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.739209697974089  
Sample 2: Real NR = 4.24, Predicted NR = 4.7944051366390195  
Sample 3: Real NR = 4.15, Predicted NR = 4.799470987769752  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.565616768602773  
Sample 5: Real NR = 4.17, Predicted NR = 4.592611019787653  
  
  
Model: KernelRidge  
Model saved as SR\_benefit\_filtered\KernelRidge\_model.pkl  
RMSE: 0.40457414738827  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.663576978285958  
Sample 2: Real NR = 4.24, Predicted NR = 4.719822471211285  
Sample 3: Real NR = 4.15, Predicted NR = 4.7256775980629  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.489647110418447  
Sample 5: Real NR = 4.17, Predicted NR = 4.515529460040922  
  
  
Model: LinearRegression  
Model saved as SR\_benefit\_filtered\LinearRegression\_model.pkl  
RMSE: 0.5153353848354232  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.957108250366676  
Sample 2: Real NR = 4.24, Predicted NR = 4.87570848994865  
Sample 3: Real NR = 4.15, Predicted NR = 4.958811035713139  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.569862729304327  
Sample 5: Real NR = 4.17, Predicted NR = 4.478317567539757  
  
  
Model: RANSACRegressor  
Model saved as SR\_benefit\_filtered\RANSACRegressor\_model.pkl  
RMSE: 0.4487855572950536  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.748707945756497  
Sample 2: Real NR = 4.24, Predicted NR = 4.795068302902906  
Sample 3: Real NR = 4.15, Predicted NR = 5.049952195354406  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.560719926244809  
Sample 5: Real NR = 4.17, Predicted NR = 4.272664492095949  
  
  
Model: TheilSenRegressor  
Model saved as SR\_benefit\_filtered\TheilSenRegressor\_model.pkl  
RMSE: 0.5105079514214927  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = 4.891843189318003  
Sample 2: Real NR = 4.24, Predicted NR = 4.884245950856252  
Sample 3: Real NR = 4.15, Predicted NR = 4.960962017644372  
Sample 4: Real NR = 5.151773101281569, Predicted NR = 4.599685364011863  
Sample 5: Real NR = 4.17, Predicted NR = 4.481020048405665  
  
  
Model: TensorFlow  
Model saved as SR\_benefit\_filtered\TensorFlow\_model.h5  
RMSE: 0.7431941166450133  
Sample predictions:  
Sample 1: Real NR = 4.65, Predicted NR = [3.8688123]  
Sample 2: Real NR = 4.24, Predicted NR = [3.9052112]  
Sample 3: Real NR = 4.15, Predicted NR = [4.563765]  
Sample 4: Real NR = 5.151773101281569, Predicted NR = [4.157097]  
Sample 5: Real NR = 4.17, Predicted NR = [4.4409657]

### Graph results

