과제 #1

- Try a **Support Vector Machine regressor** (sklearn.svm.SVR) with hyper-parameters, `C=30000, kernel="linear"`. Don't worry about what these hyperparameters mean for now.
- How does the SVR predictor perform on cross-validation (cross_val_score)?
- Try **another ML algorithm** that outperforms the above SVR predictor. And report the result as well.
  - Of course, this different ML algorithm should not be among decision tree, random forest, and linear regression.

과제 #1

- Example

```
from sklearn.svm import SVR

svm_reg = ●▲■
svm_reg.fit(housing_prepared, housing_labels)
housing_predictions = svm_reg.predict(housing_prepared)
svm_mse = mean_squared_error(housing_labels, housing_predictions)
svm_rmse = np.sqrt(svm_mse)
svm_rmse

from sklearn.model_selection import cross_val_score
# these below parameters shall be kept unchanged
svm_scores = cross_val_score(svm_reg, housing_prepared, housing_labels,
                             scoring="neg_mean_squared_error", cv=3)
svm_rmse_scores = np.sqrt(-svm_scores)
display_scores(svm_rmse_scores)
# capture the below result in the report
```

과제 #1

- Your own model

```
635  ###################################
636  ### ★★★ YOUR SECOND MODEL ★★★ ###
637  from sklearn.●▲■ import ●▲■
638
639  your_reg = ●▲■
640  your_reg.fit(housing_prepared, housing_labels)
641  your_housing_predictions = your_reg.predict(housing_prepared)
642  your_mse = mean_squared_error(housing_labels, housing_predictions)
643  your_rmse = np.sqrt(your_mse)
644  your_rmse
645
646  from sklearn.model_selection import cross_val_score
647  # these below parameters shall be kept unchanged
648  your_scores = cross_val_score(your_reg, housing_prepared, housing_labels,
649                                scoring="neg_mean_squared_error", cv=3)
650  your_rmse_scores = np.sqrt(-your_scores)
651  display_scores(your_rmse_scores)    # printing scores to check during running
652
653  ###################################
654  if your_rmse_scores.mean() < svm_rmse_scores.mean():
655      print('SUCCESS! Your RMSE Improvement: {0:0.3f}'.format(
656                          (svm_rmse_scores.mean() - your_rmse_scores.mean())))
657  else:
658      print('TRY DIFFERENTLY! Your RMSE Decrease: {0:0.3f}'.format(
659                          (svm_rmse_scores.mean() - your_rmse_scores.mean())))
660  ###################################
661  # ★★★ capture the below result in the report ★★★
```

과제 #1

```
Scores: [68813.39933699 83178.17288737 70915.38579774]
Mean: 74302.31934069823
Standard deviation: 6334.570172566263
```

Requirements:

1) Code (or ipynb) + **Report**

2) Result: Two results (SVR and the other) shall be in the form as below:

3) Runtime environment

- Versions
- Cloud or local

```
print("sklearn version", sklearn.__version__)
print("python version", sys.version)
```

```
sklearn version 1.0.2
python version 3.7.12 (default, Sep 10 2021, 00:21:48)
[GCC 7.5.0]
```

4) Summary description:

- which model you used,
- difference value of "Mean" value between two models' outputs
- any reason why you believe that your new one is better than SVR (one to several sentences),

※ 본 과제는 랭킹반영 안함

과제 #1

Submission

- Where? In LMS
- Deadline? 2 weeks later

NOTE

- Don't hesitate to contact me or main TA (Mr Baek) when facing problems in running the given source code.