OSS Project #2

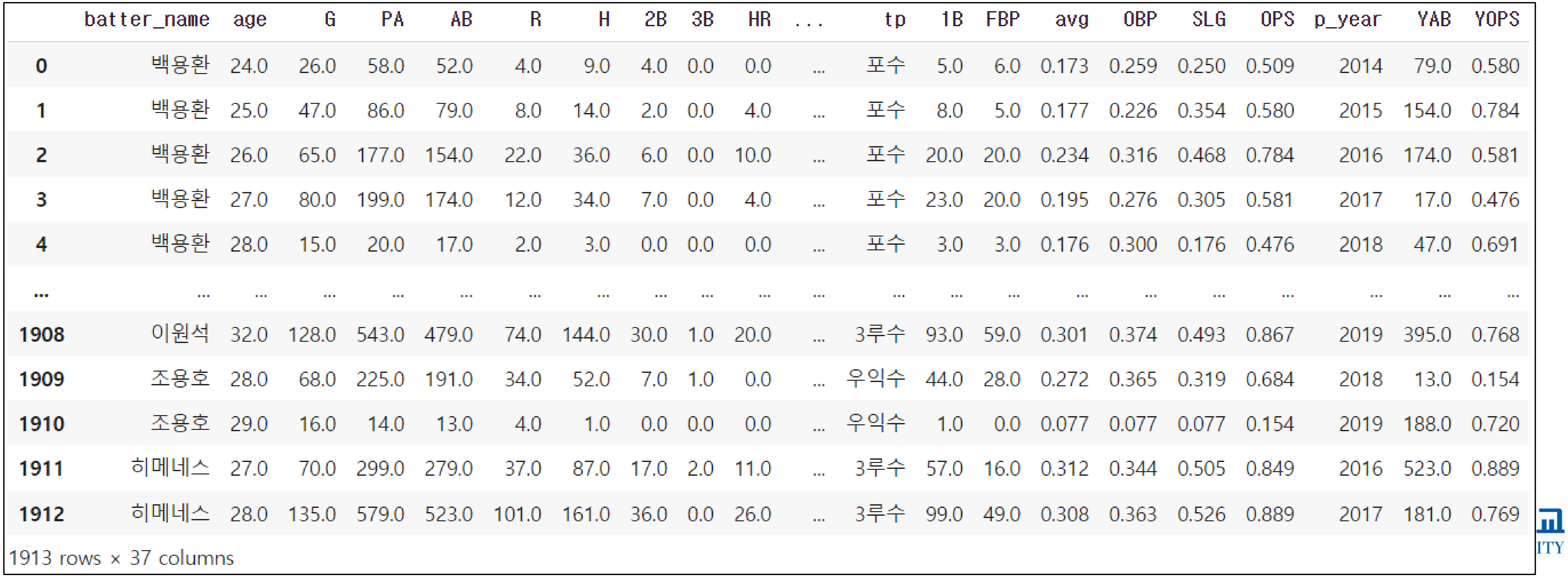
Baseball Data Analysis

Prof. Young-Duk Seo
mysid88 @inha.ac.kr

Prof. Do-Guk Kim
dgkim@inha.ac.kr

Project #2-1 Data analysis with pandas

- This data is Korean baseball Batter’s Hitting Dataset, for 1991 ~ 2018
    - Uploaded on I-Class, filename is “2019_kbo_for_kaggle_v2.csv”



	batter_name	age	G	PA	AB	R	H	2B	3B	HR	...	tp	1B	FBP	avg	OBP	SLG	OPS	p_year	YAB	YOPS
0	백용환	24.0	26.0	58.0	52.0	4.0	9.0	4.0	0.0	0.0	...	포수	5.0	6.0	0.173	0.259	0.250	0.509	2014	79.0	0.580
1	백용환	25.0	47.0	86.0	79.0	8.0	14.0	2.0	0.0	4.0	...	포수	8.0	5.0	0.177	0.226	0.354	0.580	2015	154.0	0.784
2	백용환	26.0	65.0	177.0	154.0	22.0	36.0	6.0	0.0	10.0	...	포수	20.0	20.0	0.234	0.316	0.468	0.784	2016	174.0	0.581
3	백용환	27.0	80.0	199.0	174.0	12.0	34.0	7.0	0.0	4.0	...	포수	23.0	20.0	0.195	0.276	0.305	0.581	2017	17.0	0.476
4	백용환	28.0	15.0	20.0	17.0	2.0	3.0	0.0	0.0	0.0	...	포수	3.0	3.0	0.176	0.300	0.176	0.476	2018	47.0	0.691
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1908	이원석	32.0	128.0	543.0	479.0	74.0	144.0	30.0	1.0	20.0	...	3루수	93.0	59.0	0.301	0.374	0.493	0.867	2019	395.0	0.768
1909	조용호	28.0	68.0	225.0	191.0	34.0	52.0	7.0	1.0	0.0	...	우익수	44.0	28.0	0.272	0.365	0.319	0.684	2018	13.0	0.154
1910	조용호	29.0	16.0	14.0	13.0	4.0	1.0	0.0	0.0	0.0	...	우익수	1.0	0.0	0.077	0.077	0.077	0.154	2019	188.0	0.720
1911	히메네스	27.0	70.0	299.0	279.0	37.0	87.0	17.0	2.0	11.0	...	3루수	57.0	16.0	0.312	0.344	0.505	0.849	2016	523.0	0.889
1912	히메네스	28.0	135.0	579.0	523.0	101.0	161.0	36.0	0.0	26.0	...	3루수	99.0	49.0	0.308	0.363	0.526	0.889	2017	181.0	0.769

1913 rows × 37 columns

Project #2-1 Data analysis with pandas

- Each column of the data means:
  - batter_name (선수 이름), age (나이), G (출장 경기 수), PA (타수), AB (타석 수), R (득점), H (안타), 2B (2루타), 3B (3루타), HR (홈런), TB (총 루타 수), RBI (타점), SB (도루 성공), CS (도루 실패), BB (볼넷), HBP (사구), GB (고의4구), SO (삼진), GDP (병살타), BU (희생타), fly (희생 플라이), year (해당 시즌), salary (해당 시즌의 연봉), war (승리 기여도), year_born (선수 태어난 연도), hand2 (타석위치), cp (최근 포지션), tp (통합포지션), 1B (1루타), FBP (BB + HBP), avg (타율), OBP (출루율), SLG (장타율), OPS (OBP + SLG), p_year (다음시즌), YAB (다음 시즌 타석 수), YOPS (다음 시즌 OPS)

Project #2-1 Data analysis with pandas

- Project Requirements **(A total of 45 points)**
  - **Please implement the Python source code corresponding to the below requirements**

1) Print the top 10 players in hits (안타, H), batting average (타율, avg), homerun (홈런, HR), and on-base percentage (출루율, OBP) for each year from 2015 to 2018. **(15 points)**

2) Print the player with the highest war (승리 기여도) by position (cp) in 2018. **(15 points)**
   - Position info. - 포수, 1루수, 2루수, 3루수, 유격수, 좌익수, 중견수, 우익수

3) Among R (득점), H (안타), HR (홈런), RBI (타점), SB (도루), war (승리 기여도), avg (타율), OBP (출루율), and SLG (장타율), which has the highest correlation with salary (연봉)? **(15 points)**
   - Implement code to calculate correlations and print the answer to the above question.

Project #2-2 Data analysis with sklearn

- Project Goal
  - Train various ML models to predict the salary of the batter in the specific year

  - This is a regression task and we will use three kinds of ML models
    - Decision Tree Regressor
    - Random Forest Regressor
    - Support Vector Machine Regressor

  - We will use only numerical features

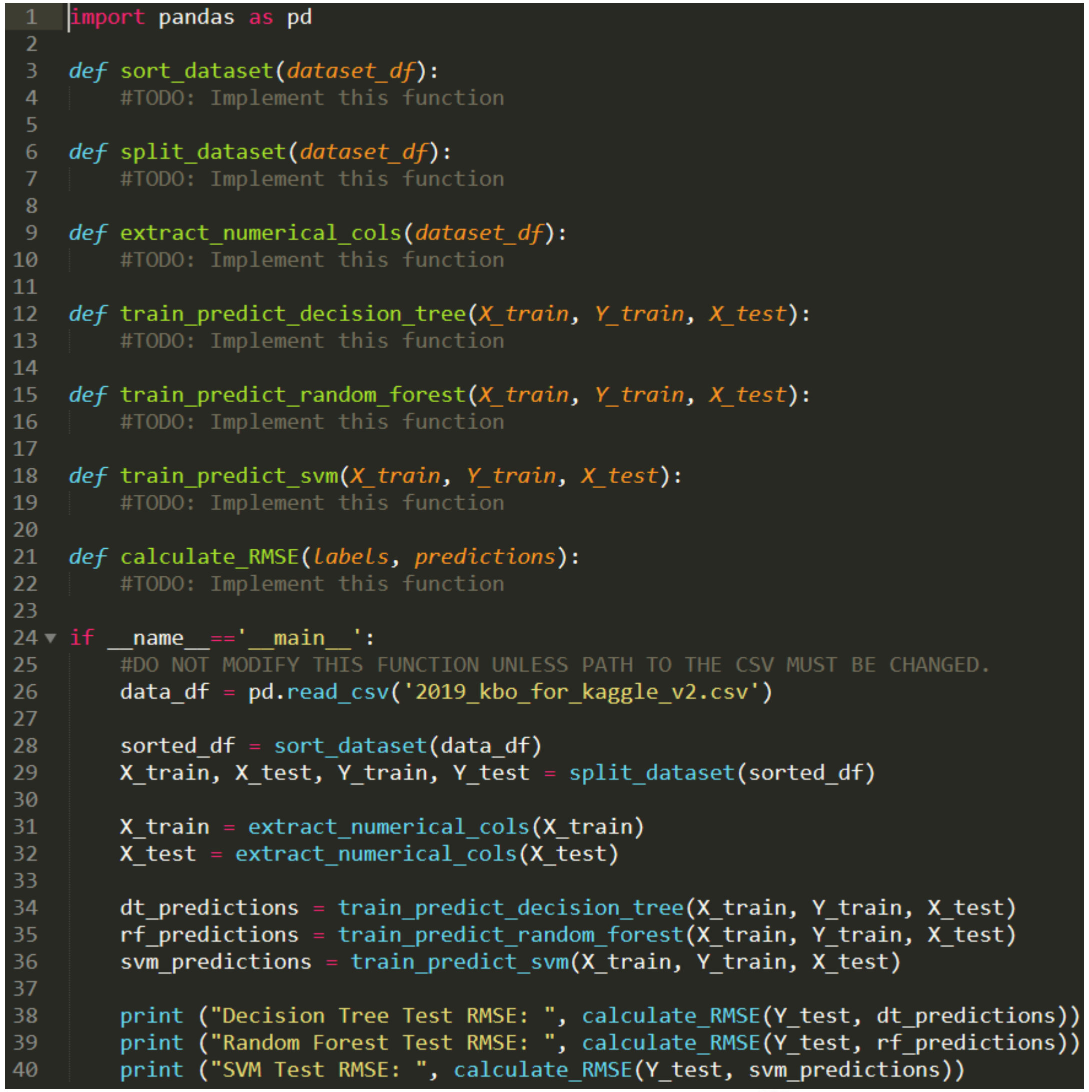
Project #2-2 Data analysis with sklearn

- Project Requirements **(A total of 45 points)**
  - **Using same data with Project #2-1, please implement source code that satisfies below requirements**

1) Sort the entire data by year(해당 시즌) column in ascending order **(7 points)**
2) Split the entire data as train/test datasets **(10 points)**
3) Extract only numerical columns **(7 points)**
   - Numerical columns: `'age'`, `'G'`, `'PA'`, `'AB'`, `'R'`, `'H'`, `'2B'`, `'3B'`, `'HR'`, `'RBI'`, `'SB'`, `'CS'`, `'BB'`, `'HBP'`, `'SO'`, `'GDP'`, `'fly'`, `'war’`
4) Complete the train and predict functions for decision tree, random forest and svm **(15 points)**
5) Calculate RMSE for given labels and predictions **(6 points)**

Project #2-2 Data analysis with sklearn

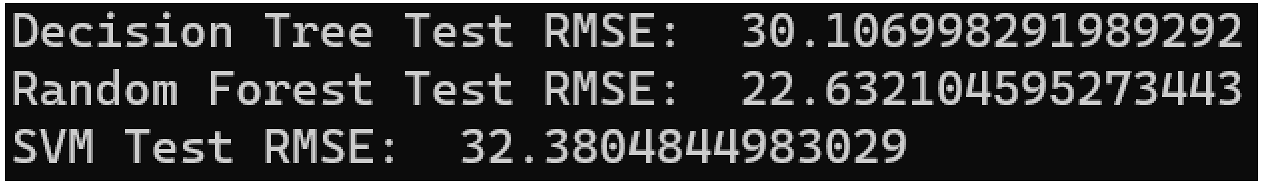
- Project Code Template
  - The template for the code is provided on the I-Class and you must implement functions in the template and submit the completed code
  - You can import additional modules that you need to implement each function
  - Do not modify the function header (function name and parameter names)



```python
import pandas as pd

def sort_dataset(dataset_df):
    #TODO: Implement this function

def split_dataset(dataset_df):
    #TODO: Implement this function

def extract_numerical_cols(dataset_df):
    #TODO: Implement this function

def train_predict_decision_tree(X_train, Y_train, X_test):
    #TODO: Implement this function

def train_predict_random_forest(X_train, Y_train, X_test):
    #TODO: Implement this function

def train_predict_svm(X_train, Y_train, X_test):
    #TODO: Implement this function

def calculate_RMSE(labels, predictions):
    #TODO: Implement this function

if __name__=='__main__':
    #DO NOT MODIFY THIS FUNCTION UNLESS PATH TO THE CSV MUST BE CHANGED.
    data_df = pd.read_csv('2019_kbo_for_kaggle_v2.csv')

    sorted_df = sort_dataset(data_df)
    X_train, X_test, Y_train, Y_test = split_dataset(sorted_df)

    X_train = extract_numerical_cols(X_train)
    X_test = extract_numerical_cols(X_test)

    dt_predictions = train_predict_decision_tree(X_train, Y_train, X_test)
    rf_predictions = train_predict_random_forest(X_train, Y_train, X_test)
    svm_predictions = train_predict_svm(X_train, Y_train, X_test)

    print ("Decision Tree Test RMSE: ", calculate_RMSE(Y_test, dt_predictions))
    print ("Random Forest Test RMSE: ", calculate_RMSE(Y_test, rf_predictions))
    print ("SVM Test RMSE: ", calculate_RMSE(Y_test, svm_predictions))
```

Project #2-2 Data analysis with sklearn

- Example Result
    - After completing the implementation and running the source code with the CSV file in the same folder with the source code, you can get a result similar to the one below



```
Decision Tree Test RMSE:  30.106998291989292
Random Forest Test RMSE:  22.632104595273443
SVM Test RMSE:  32.3804844983029
```

Project #2-2 Data analysis with sklearn

- Function descriptions
  - **sort_dataset**
    - Return sorted version of the given dataframe by year(해당 시즌) column in ascending order
  - **split_dataset**
    - Return X_train, X_test, Y_train, Y_test dataframes
    - We use the salary column for the label
      - **Please rescale label value through multiply by 0.001**
    - Split the index range of [:1718] for the given dataframe as the train dataset
    - Split the index range of [1718:] for the given dataframe as the test dataset

Project #2-2 Data analysis with sklearn

- Function descriptions
  - **extract_numerical_cols**
    - Return a dataframe that extracts only numerical features from the input dataframe
    - Numerical columns: 'age', 'G', 'PA', 'AB', 'R', 'H', '2B', '3B', 'HR', 'RBI', 'SB', 'CS', 'BB', 'HBP', 'SO', 'GDP', 'fly', 'war’
  - **train_predict_decision_tree**
    - Train decision tree regressor model using given X_train and Y_train
    - Return prediction result of X_test by using the trained model

Project #2-2 Data analysis with sklearn

- Function descriptions
  - **train_predict_random_forest**
    - Train random forest regressor model using given X_train and Y_train
    - Return prediction result of X_test by using the trained model
  - **train_predict_svm**
    - Train **the pipeline consists of a standard scaler and SVM** model using given X_train and Y_train
    - Return prediction result of X_test by using the trained model

Project #2-2 Data analysis with sklearn

- Function descriptions
  - **calculate_RMSE**
    - Calculate and return RMSE using given labels and predictions

Project #2 Submission

- Submission requires two python source code file
  ① Source code for Project #2-1
  ② Source code for Project #2-2

- Please submit your file on the I-Class
- Due date is 12/3(Sun) 23:59

- TA will verify your submissions using another auto-script and copy-checking tools

Project #2 Evaluation

- GitHub upload **(10 points)**
  - Please upload your Project #2 source codes on GitHub in period of 12/4(Mon) 00:00~ 12/6(Wed) 23:59 and submit repository URL on I-Class
  - **Don’t upload your source codes on GitHub before 12/4(Mon)!**

- Therefore, Project #2 score will be evaluated as:
  - Project #2-1(45 points) + Project #2-2(45 points) + GitHub Upload(10 points)
  - A total of 100 points

- If you have any questions, don’t hesitate to post questions on I-Class Q&A