# CS 6140 Machine Learning: Homework 1

Hongyang R. Zhang

Due 2023/09/27 11:59pm

**Instructions:**

- Each student should write their solution independently and submit a single PDF file that includes all the solutions on Gradescope. The running codes, such as jupyter notebooks, can be submitted as supplementary materials on Canvas. Include the names of classmates you discussed with at the beginning of the solution.
- There are up to three late days for all the problem sets and project submissions. Use them wisely. Late submissions are considered case by case. Please reach out to us if you need extra late days.

**Instructions for Python** To download `Python` packages, use

```
pip install packageName
```

To use the installed package, type

```
import packageName
```

**Problem 1** [12pts]
For each of parts (a) through (c), indicate whether we would generally expect the performance of a flexible statistical learning method to be better or worse than an inflexible method. Justify your answer.

(a) [4pts] The sample size $n$ is extremely large, and the number of predictors $p$ is small.

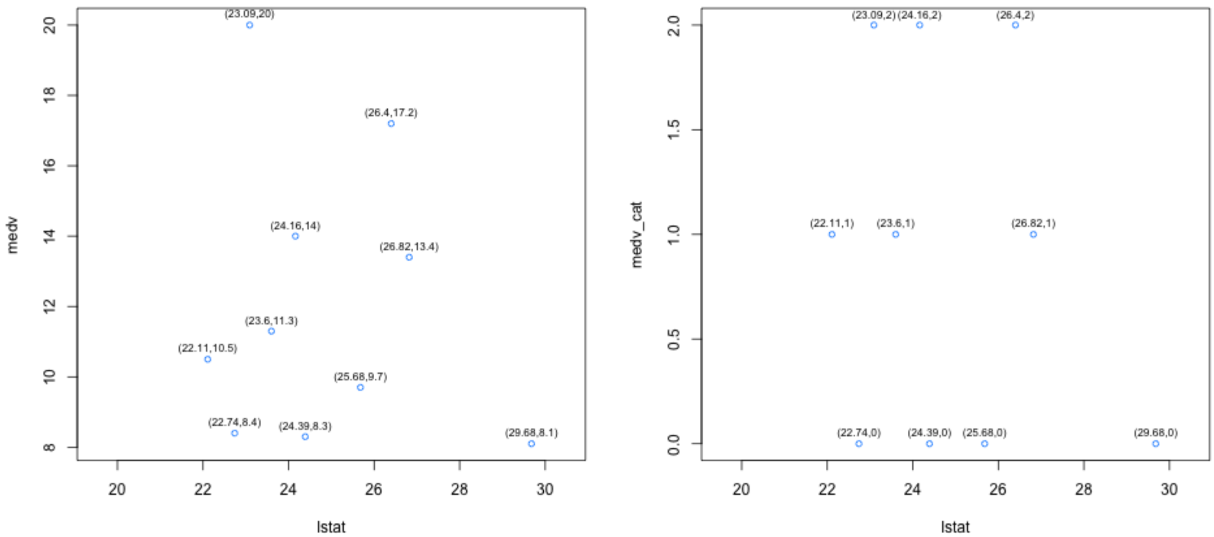
(b) [4pts] The number of predictors $p$ is extremely large, and the number of observations $n$ is small.

(c) [4pts] The relationship between the predictors and response is highly non-linear.

**Problem 2** [27pts]

For this problem, we are going to use KNN to predict the house value based on a small sample of the Boston housing data shown in Figure 1.

| | |
|---|---|
| `lstat` | percentage of households with low socioeconomic status |
| `medv` | median house value in \$1,000's (continuous variable) |
| `medv_cat` | category of median house value (0: low; 1: medium; 2: high) |




(a) regression problem (b) classification problem

Figure 1: A sample of Boston housing data

(a) [5pts] Use Figure 1a to predict `medv` given `lstat=25` with $K = 1$ and $K = 5$.

(b) [4pts] Repeat (a) for `lstat=27`.

(c) [5pts] Use Figure 1b to predict `medv_cat` given `lstat=25` with $K = 1$ and $K = 5$.

(d) [4pts] Repeat (c) for `lstat=27`.

(e) [4pts] If we increase $K$ in KNN, is the model more flexible or less flexible? Explain why.

(f) [5pts] How do the square of bias, variance, training MSE, test MSE, and irreducible error change with $K$ for KNN regression? Explain why.

**Problem 3** [37pts]
In this problem, we are going to use simulated data sets to understand better how the square of bias, variance, irreducible error, and MSE vary with model flexibility.

(a) [4pts] Generate a simulated data set as follows:

```
def f(x):
    return x ** 5 - 2 * x ** 4 + x ** 3

def get_sim_data(f, sample_size=100, std=0.01):
    x = np.random.uniform(0, 1, sample_size)
    y = f(x) + np.random.normal(0, std, sample_size)
    df = pd.DataFrame({'x': x, 'y': y})
    return df
```

In this data set, what is the number of observations $n$ and what is the number of features $p$ (different powers of $x$ are counted as different features)? Write out the model used to generate the data in equation form.

(b) [5pts] Fit the polynomial functions of a degree from 0 to 15 using the simulated data in (a):

$$
\begin{aligned}
&f_0(x) = \beta_0 + \varepsilon \\
&f_1(x) = \beta_0 + \beta_1 x + \varepsilon \\
&f_2(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon \\
&\vdots \\
&f_{15}(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 \cdots + \beta_{15} x^{15} + \varepsilon
\end{aligned}
$$

(Hint: You may find

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
```

useful.)

(c) [4pts] Predict the response at $x_0 = 0.18$ using the fitted functions in (b).

(d) [4pts] Repeat (a)-(c) for 250 times.

(e) [4pts] Use (d) to calculate the square of bias for the fitted polynomials $\hat{f}_0(x_0), \hat{f}_1(x_0), \cdots, \hat{f}_{15}(x_0)$.

(f) [4pts] Use (d) to calculate the variance for the fitted polynomials $\hat{f}_0(x_0), \hat{f}_1(x_0), \cdots, \hat{f}_{15}(x_0)$.

(g) [4pts] Calculate the irreducible error based on the data-generating process.

(h) [4pts] Calculate the MSE based on (e), (f), and (g).

(i) [4pts] Plot how the square of bias, variance, irreducible error, and MSE vary with the degree of polynomials. Explain your findings.

**Problem 4** [24pts]
We will now perform cross-validation on a simulated data set.

(a) [4pts] Generate a simulated data set as follows:

```
def f(x):
    return x ** 5 - 2 * x ** 4 + x ** 3

np.random.seed(1)
x = np.random.uniform(0, 1, size=500)
y = f(x) + np.random.normal(0, 0.01, size=500)
```

(b) [4pts] Create a scatterplot of $x$ against $y$. Comment on what you find. (Hint: You may find `plot()` helpful)

(c) [4pts] Set a random seed `123`, and then compute the LOOCV errors that result from fitting the polynomial functions of degree from 1 to 7 using the simulated data in (a):

$$f_1(x) = \beta_0 + \beta_1 x + \varepsilon$$
$$f_2(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon$$
$$\vdots$$
$$f_7(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 \cdots + \beta_7 x^7 + \varepsilon$$

(Hint: See Section 5.3 in ISLP for an example of how to implement cross-validation in Python. You may find

```
from sklearn.model_selection import cross_validate
from ISLP.models import sklearn_sm
```

helpful )

(d) [4pts] Repeat (c) using another random seed `12345`, and report your results. Are your results the same as what you got in (c)? Why?

(e) [4pts] Which of the models in (c) had the smallest LOOCV error? Is this what you expected? Explain your answer.

(f) [4pts] Fit $f_5(x)$ using least squares. Comment on the coefficient estimates and their statistical significance.