TU: Linear Regression

Industrial AI & Automation by Y.K.Kim

Mod: 2024-2

Author: Jin Kwak/21900031

Date: 24.09.20

Introduction

Fit Linear Regression

Include path

```
addpath('../../Library')
```

Examples

Example 1: Fit with Linear Regression

Data Acquisition

- Feature: One-Dimension, p=1
- True value: ytrue = 2X+3

```
x = randn(100,1);
y = x*2 + 3+randn(100,1);
```

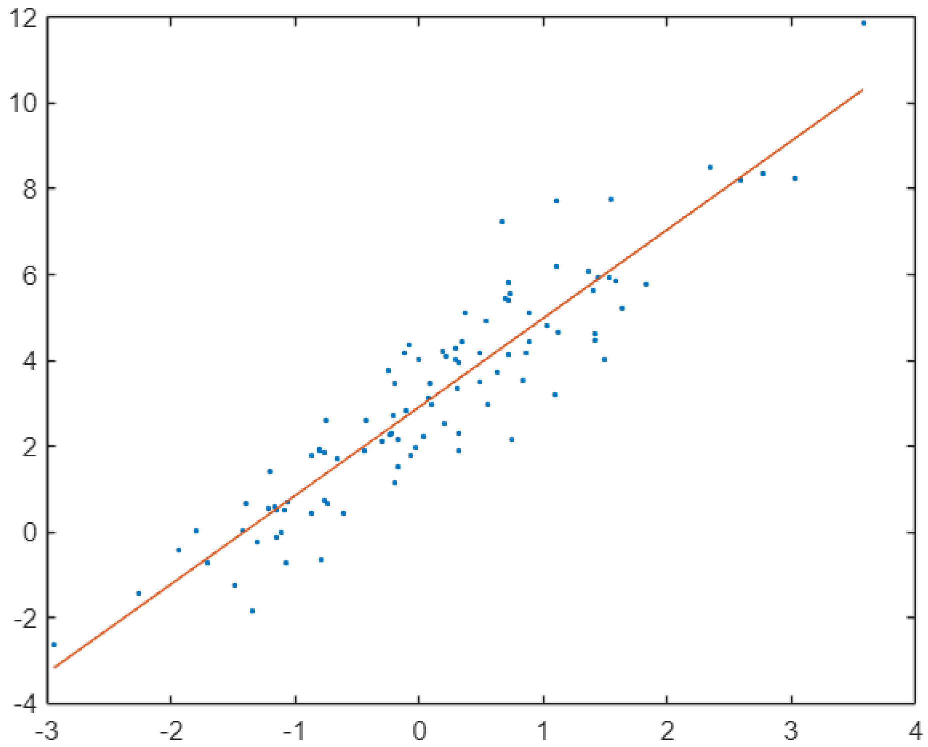
Fit Linear Regression

```
mdl = fitlm(x,y)
```

```
mdl = 
선형 회귀 모델:
    y ~ 1 + x1

추정된 계수:
                   Estimate       SE        tStat       pValue  
                   ________    ________    ______    __________

    (Intercept)     2.9193       0.1013    28.819    1.1939e-49
    x1              2.0652     0.087093    23.713    2.1985e-42


관측값 개수: 100, 오차 자유도: 98
RMS 오차: 1.01
결정계수: 0.852, 수정된 결정계수: 0.85
상수 모델에 대한 F-통계량: 562, p-값 = 2.2e-42
```

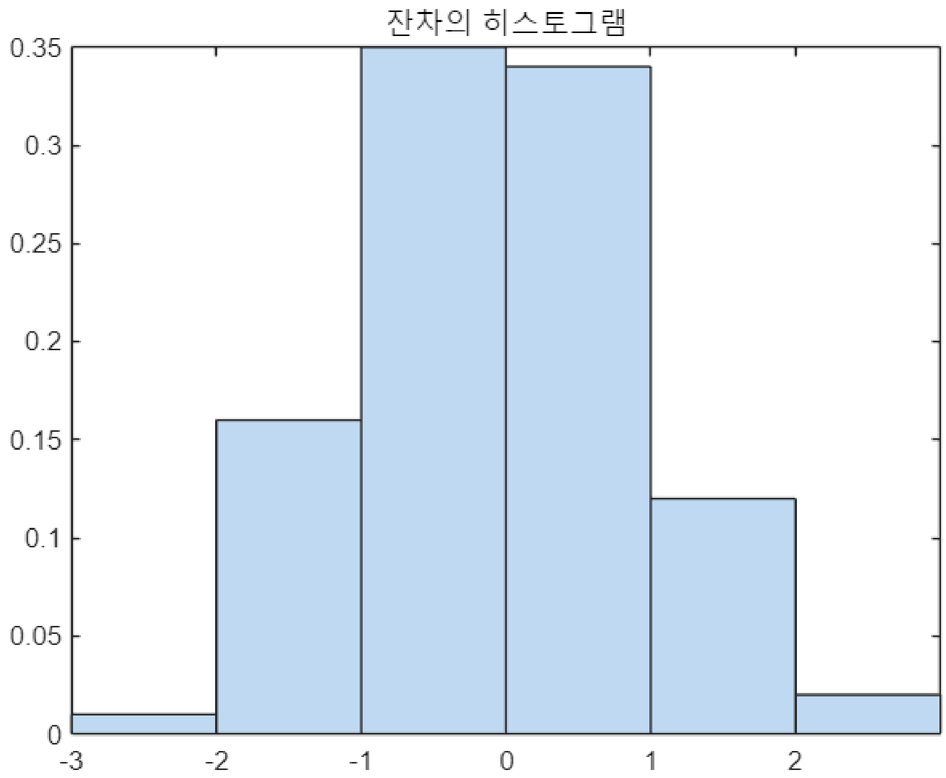
```

% Plot prediction
```

```
plot(x,y,'.',x,mdl.Fitted, '-')
```



Analyze Linear Regression

```
% Plot residual histogram
plotResiduals(mdl)
```



```
% From remove outlier from histogram
outl = find(abs(mdl.Residuals.Raw) > 2);
mdl.Residuals.Raw(outl)
```

```
ans = 3×1
    2.9449
    2.5349
   -2.2979
```

```
% Fit with outlier removed
mdl2 = fitlm(x,y,'Exclude',outl)
```

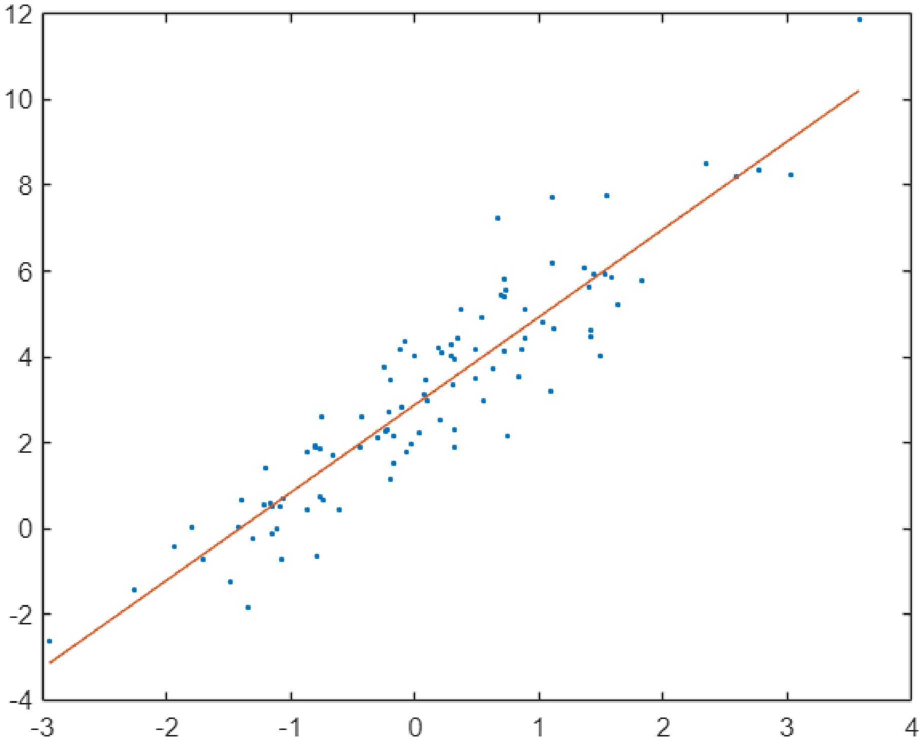
```
mdl2 = 
선형 회귀 모델:
    y ~ 1 + x1

추정된 계수:
                   Estimate       SE        tStat       pValue  
                   ________    ________    ______    __________

    (Intercept)     2.8886     0.092874    31.102    1.3369e-51
    x1              2.0446     0.079297    25.784    1.1196e-44


관측값 개수: 97, 오차 자유도: 95
RMS 오차: 0.911
결정계수: 0.875, 수정된 결정계수: 0.874
상수 모델에 대한 F-통계량: 665, p-값 = 1.12e-44
```

Predict with Test data

```
% Ypredict and confidence interval
Xnew=2;
[Ynew,YnewI]=predict(mdl2,Xnew)
```

```
Ynew = 6.9777
YnewI = 1×2
    6.6268    7.3286
```

```

% Plot prediction
plot(x,y,'.',x,mdl2.Fitted, '-')
```



Example 2: Predict Car MPG

Find linear relationship of MPG(연비) with Weight & Displacement

Then, Predict MPG for a car with Weight=3000 , Displacement=300

Data Acquisition

- Dependet Variable: MPG

- Independet Variables: Weight, Displacement

```
clear
load carsmall

tbl = table(MPG,Weight, Displacement);  % table type
```

Fit Linear Regression

```
mdl = fitlm(tbl,'MPG~Weight+Displacement')
```

```
mdl = 
선형 회귀 모델:
    MPG ~ 1 + Weight + Displacement

추정된 계수:
                     Estimate        SE         tStat       pValue  
                    __________    _________    _______    __________

    (Intercept)         46.925       2.0858     22.497    6.0509e-39
    Weight          -0.0068422    0.0011337    -6.0353    3.3838e-08
    Displacement     -0.014593    0.0082695    -1.7647      0.080968


관측값 개수: 94, 오차 자유도: 91
RMS 오차: 4.09
결정계수: 0.747, 수정된 결정계수: 0.741
상수 모델에 대한 F-통계량: 134, p-값 = 7.22e-28
```

```
mdl.CoefficientNames
```

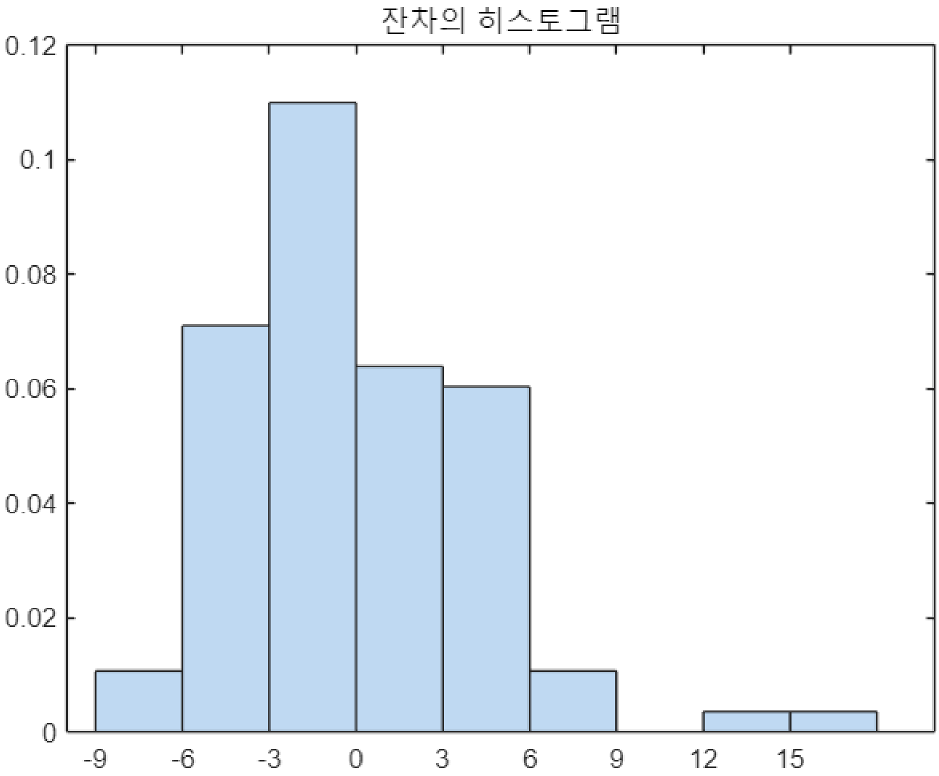
```
ans = 1×3 cell
'(Intercept)''Weight'     'Displacement'
```

```
mdl.Coefficients.Estimate
```

```
ans = 3×1
   46.9247
   -0.0068
   -0.0146
```

Analyze Linear Regression

```
plotResiduals(mdl)
```



```
% Remove outlier
outl = find((mdl.Residuals.Raw) > 9);

% Fit with outlier removed
mdl2 = fitlm(tbl,'MPG~Weight+Displacement','Exclude',outl)
```

```
mdl2 = 
선형 회귀 모델:
    MPG ~ 1 + Weight + Displacement

추정된 계수:
                     Estimate        SE         tStat       pValue  
                    __________    __________    _______    __________

    (Intercept)         45.548        1.8056     25.226    2.5559e-42
    Weight          -0.0062503    0.00097845    -6.3879    7.4767e-09
    Displacement     -0.018035     0.0071414    -2.5255      0.013324


관측값 개수: 92, 오차 자유도: 89
RMS 오차: 3.51
결정계수: 0.796, 수정된 결정계수: 0.792
상수 모델에 대한 F-통계량: 174, p-값 = 1.78e-31
```

```
mdl2.Coefficients.Estimate
```

```
ans = 3×1
   45.5477
   -0.0063
   -0.0180
```

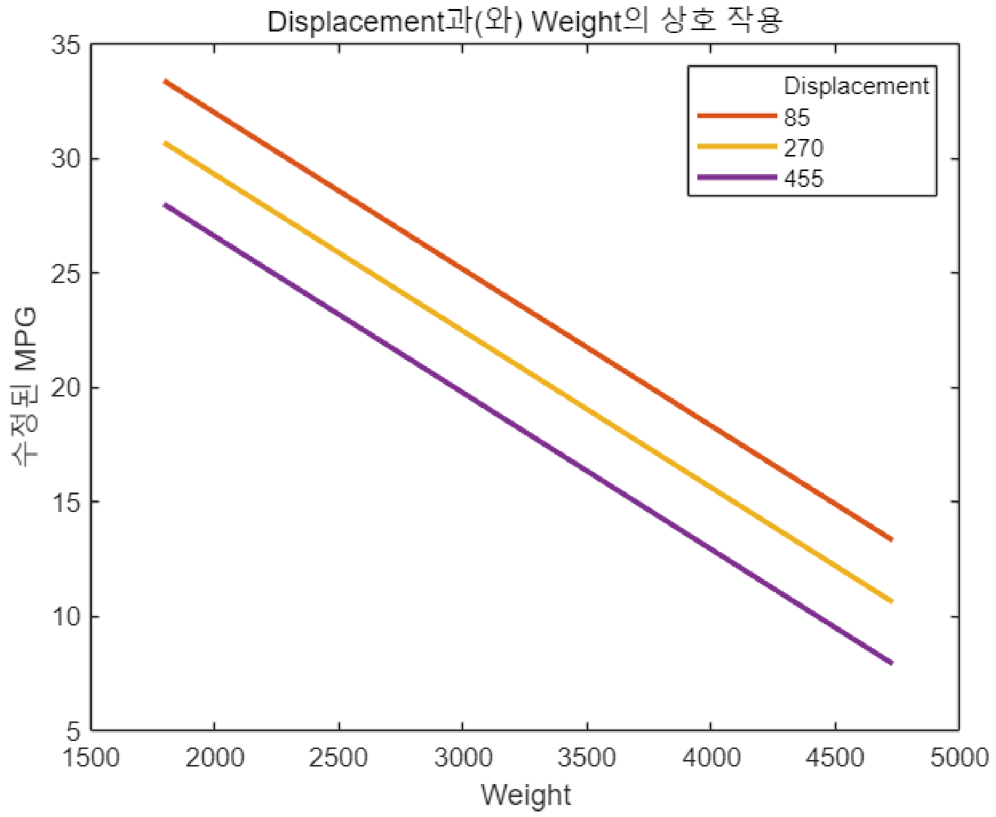
Predict

Detailed look at the interactions

```
% Ypredict and confidence interval
Xnew=[3000 300];
[Ynew,YnewI]=predict(mdl2,Xnew)
```

```
Ynew = 21.3863
YnewI = 1×2
   19.8515   22.9211
```

```
plotInteraction(mdl,'Displacement','Weight','predictions')
```



Exercise

Exercise 1 : Gradient Descent

Linear Regression Using Gradient Descent

Hypothesis.

$$h_\theta(x) = \theta_1 x + \theta_0$$

To find the parameters, repeat until convergence

$$\theta_k = \theta_k - \alpha \frac{\partial}{\partial \theta_k} J(\theta_1, \theta_0)$$

where, cost(error) function

$$J(\theta_1, \theta_0) = \frac{1}{2n} E = \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - h(x_i) \right)^2$$

and

$$\frac{\partial J}{\partial \theta_1} = -\frac{1}{n} \sum_{i=1}^{n} x_i \left(y_i - (\theta_1 x_i + \theta_0) \right)$$

$$\frac{\partial J}{\partial \theta_0} = -\frac{1}{n} \sum_{i=1}^{n} \left(y_i - (\theta_1 x_i + \theta_0) \right)$$

Data Acquisition

Feature: One-Dimension, p=1

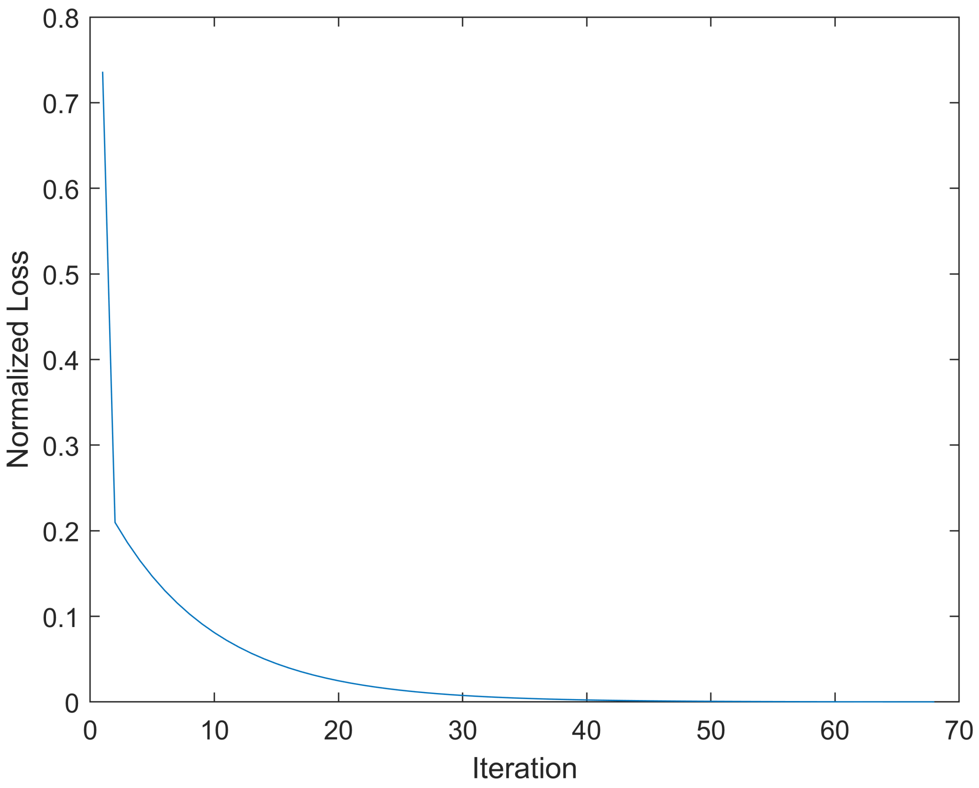
True: y = 2X+3

```
N=100;
X = randn(N,1);
Y = X*2 + 3+randn(N,1);
```

Fit Linear Regression: Gradient Descent

```
% dJ/dx
lambda=0.1;  % learning rate

% Initialization for t0, t1
t0=0.5;
t1=0.5;
loss=1;
itrN=1000;
k=1;

t1_prev=0;
t0_prev=0;
```

```
%% ADD your code here
while(loss>0.0001 && k<itrN)

    % Define function h(x)
    h= t1*X + t0;

    % Define gradient w.r.t theta_1 and theta_0
    dJt1 = - 1/N * sum(X .* (Y- (t1*X + t0)));
    dJt0 = -1/N * sum(Y- (t1*X + t0));

    % Update theta1, theta0
    t1= t1 - lambda * dJt1;
    t0= t0 - lambda * dJt0;

    % Cost Function v1
%     loss=sum((Y-h).^2)/(2*N);

    % Cost Function v2
    loss= (norm(t1-t1_prev)+norm(t0-t0_prev))/2;
    t1_prev=t1;
    t0_prev=t0;

    loss_hist(k) = loss;
    k=k+1;
end


% Plot loss vs iteration
figure;
plot(loss_hist)
xlabel('Iteration');
ylabel('Normalized Loss');
```



```
% Predicted hypothesis y(x)
ypred= t1*X + t0 ;

disp('optimal paraterms are')
```

```
optimal paraterms are
```

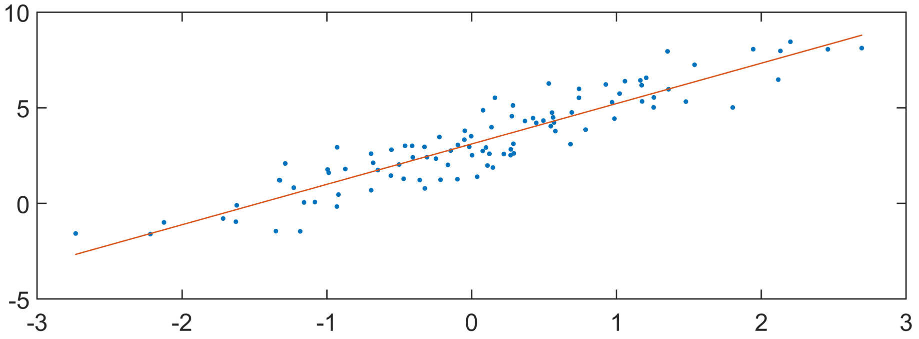
```
t1
```

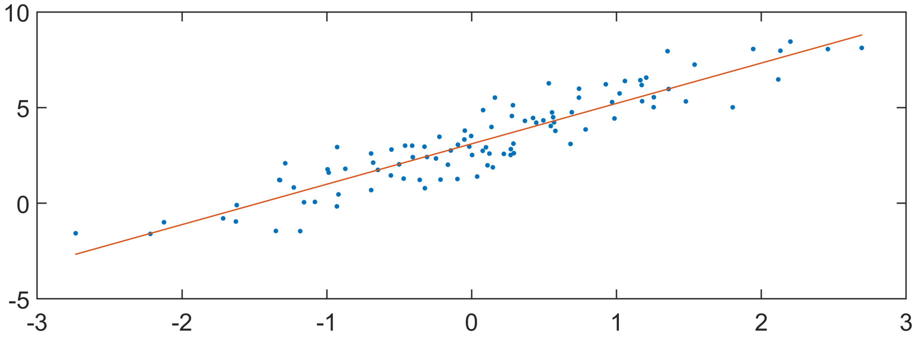
```
t1 = 2.1112
```

```
t0
```

```
t0 = 3.1064
```

```
% MATLAB fit linear using fitlm(x,y)
mdl = fitlm(X,Y);  %%% ADD your code here

% Plot and compare predictions
figure
subplot(2,1,1)
plot(X,Y,'.',X,mdl.Fitted, '-')
subplot(2,1,2)
plot(X,Y,'.',X,ypred, '-')
```





Exercise 2 : Linear Regression with dim=2

Find the linear regression. Remove Outlier and predict a test value Xtest=[2;1];. You can use MATLAB fitlm()

- Feature: 2-Dimensions, p=2
- True value: y = 2*X1+4*X2+3

Data Acquisition

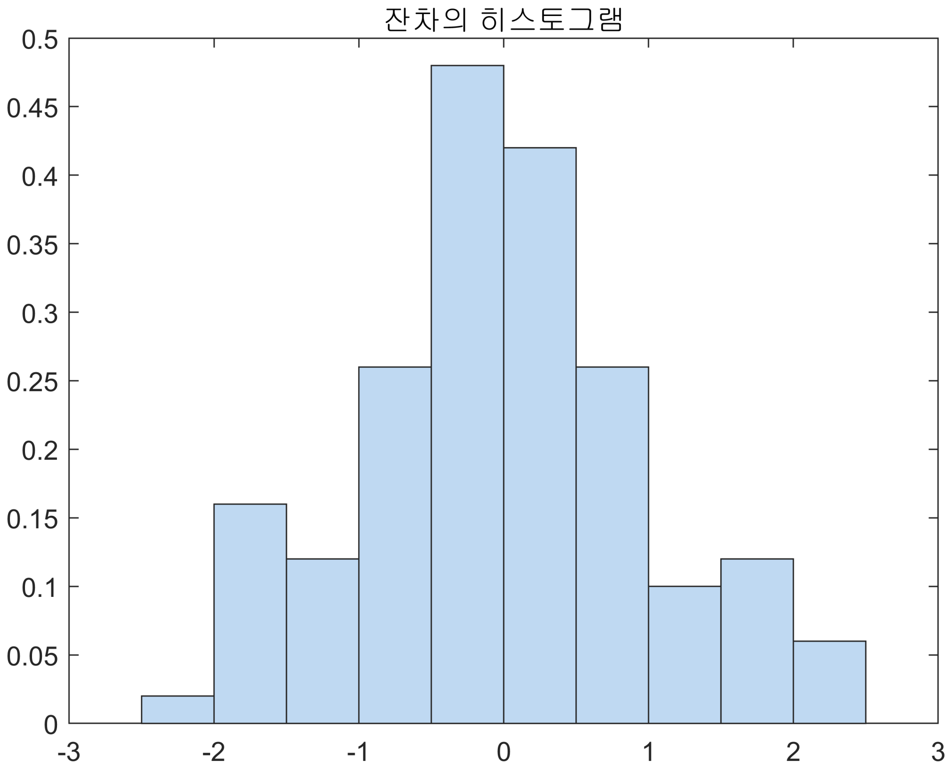
```
X = randn(100,2);
y = X*[2;4] + 3+ randn(100,1);
```

Fit Linear Regression

```
mdl= fitlm(X,y);
```

Analyze

```
% Remove outlier
%  -plot residual histogram
figure
plotResiduals(mdl)
```

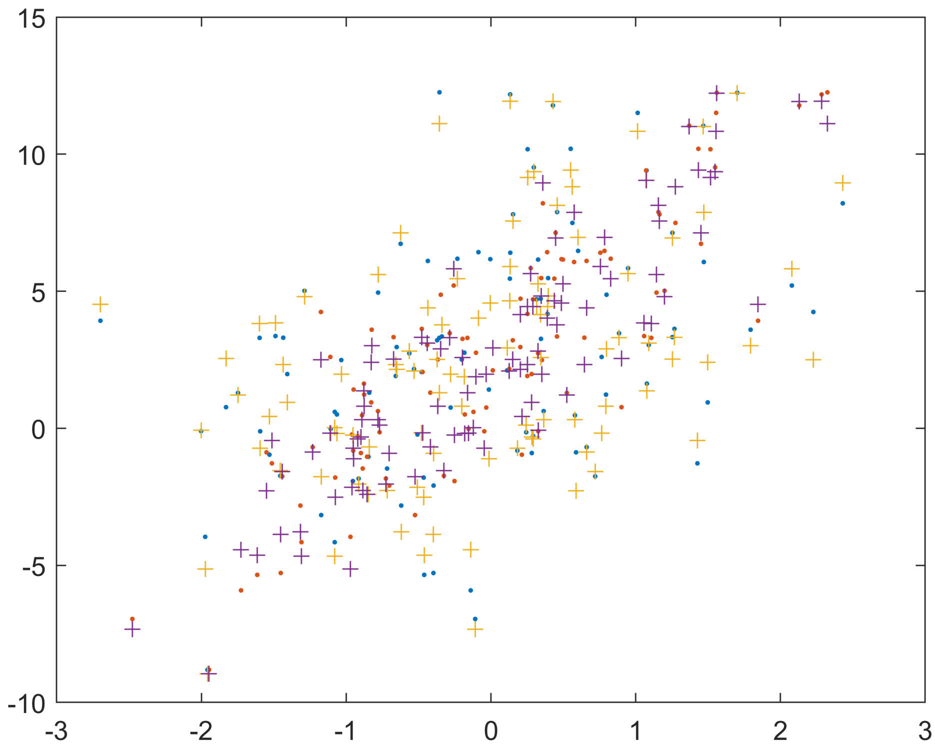


```
% - remove outlier from histogram analysis
outl = find((mdl.Residuals.Raw) > 1);

% Fit linear regression (fitlm)
mdl2 = fitlm(X,y,'Exclude',outl);
```

Predict

Predict for a Test value

```
Xnew=[2, 1];
[Ynew, YnewI]= predict(mdl2,Xnew);
plot(X,y,'.', X,mdl2.Fitted,'+')
```



Exercise 3: Linear Regression with dim=4

Find linear relationship of

- MPG ~ 1 + Acceleration + Displacement + Horsepower + Weight

Choose a test data and predict

Data Acquisition

```
clear
load carsmall

tbl = table(MPG,Acceleration,Displacement,Horsepower,Weight);
```

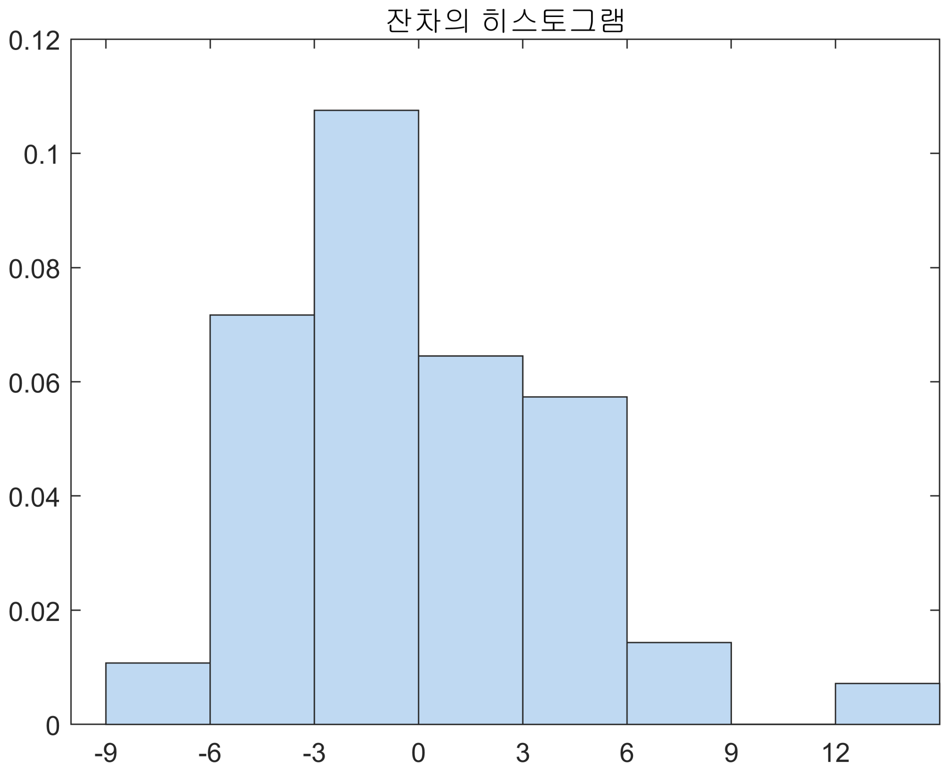
Fit Linear Regression

```
mdl = fitlm(tbl, 'MPG~Acceleration+Displacement+Horsepower+Weight')
```

```
mdl = 
선형 회귀 모델:
    MPG ~ 1 + Acceleration + Displacement + Horsepower + Weight

추정된 계수:
                     Estimate        SE          tStat        pValue
```

```
                        __________    _________    ________    __________

    (Intercept)             48.117       3.9008      12.335    6.9194e-21
    Acceleration         -0.060312      0.21167    -0.28493       0.77636
    Displacement        -0.0066826     0.011594    -0.57638       0.56583
    Horsepower           -0.037547     0.026139     -1.4364       0.15442
    Weight               -0.006084    0.0013823     -4.4014      3.01e-05


관측값 개수: 93, 오차 자유도: 88
RMS 오차: 4.11
결정계수: 0.753, 수정된 결정계수: 0.742
상수 모델에 대한 F-통계량: 67.1, p-값 = 6.49e-26
```

```
plotResiduals(mdl);
```



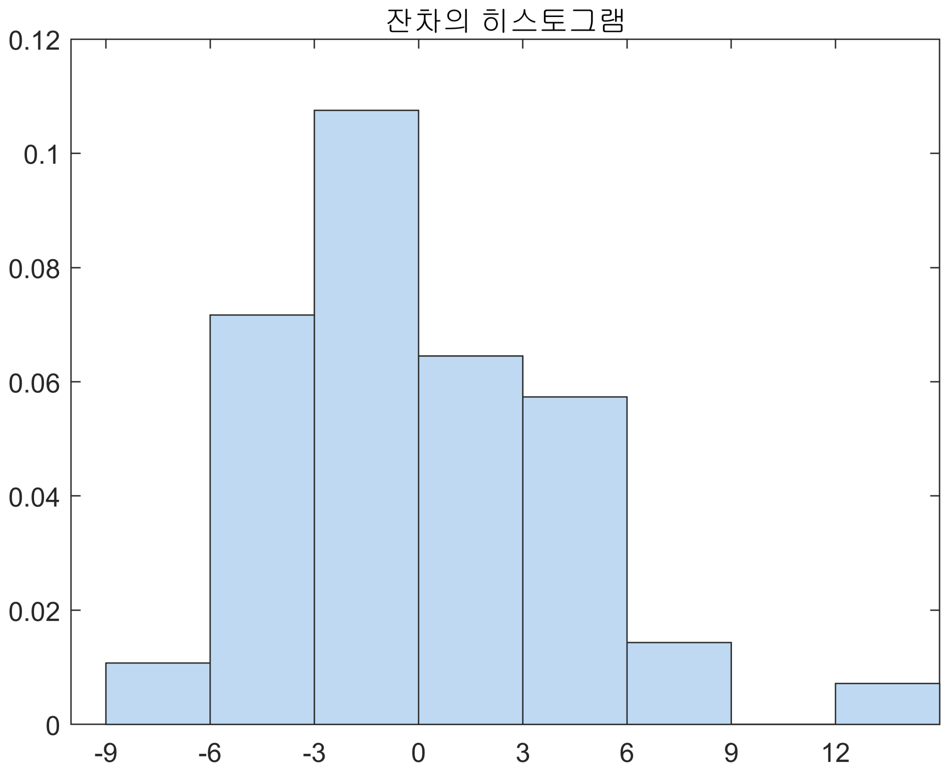
Analyze

```
% Remove outlier
%  -plot residual histogram
% - remove outlier from histogram analysis
outl =find((mdl.Residuals.Raw) > 9); % find from Residuals.Raw
mdl.Residuals.Raw(outl)
```

```
ans = 2×1
   14.1936
   12.9259
```

```
% Fit linear regression (fitlm)
```

```
mdl2 = fitlm(tbl,'Exclude',outl);

plotResiduals(mdl);
```



```
mdl2.Coefficients.Estimate
```

```
ans = 5×1
10^3 ×
    1.5188
   -0.0326
    0.0573
    0.0034
    0.0060
```

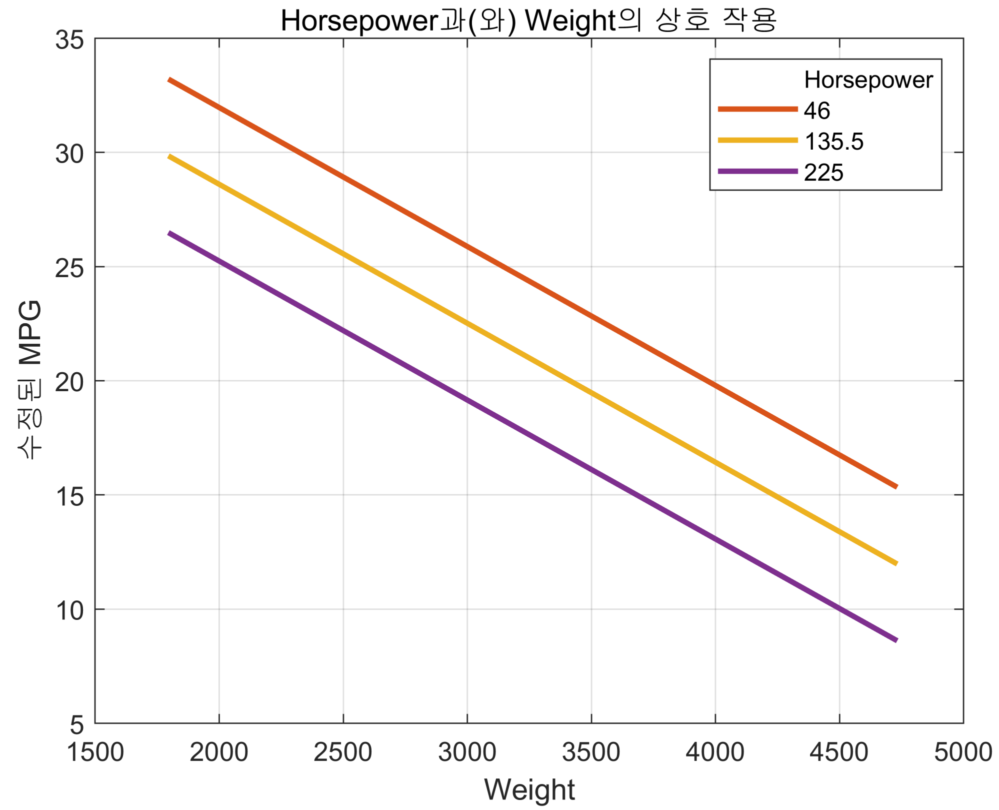
```
mdl2.Coefficients.SE
```

```
ans = 5×1
  523.0048
    8.2575
   15.3438
    0.8172
    1.7691
```

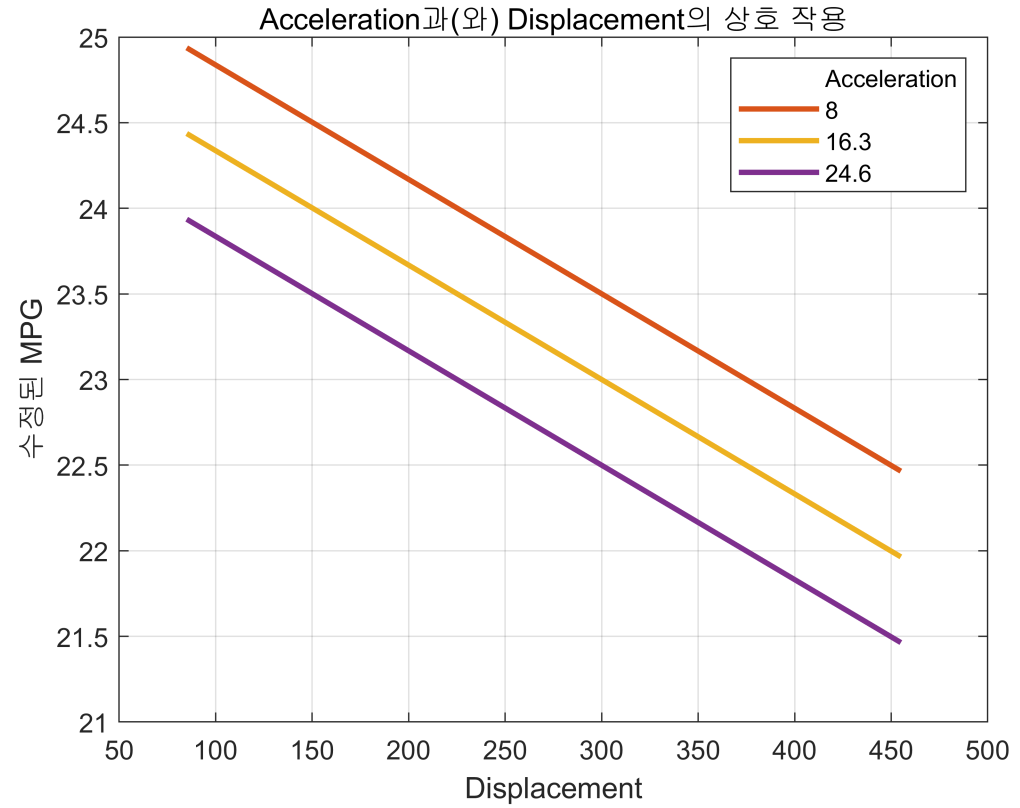
Predict

Predict for a Test value

```
Xnew = [10 300 150 3500];
[Ynew,YnewI]=predict(mdl2,Xnew);
figure, clf, box on; hold on; grid on;
plotInteraction(mdl,'Horsepower','Weight','predictions')
```



```
figure, clf, box on; hold on; grid on;
plotInteraction(mdl,'Acceleration','Displacement','predictions')
```


Acceleration과(와) Displacement의 상호 작용
수정된 MPG
Displacement
Acceleration