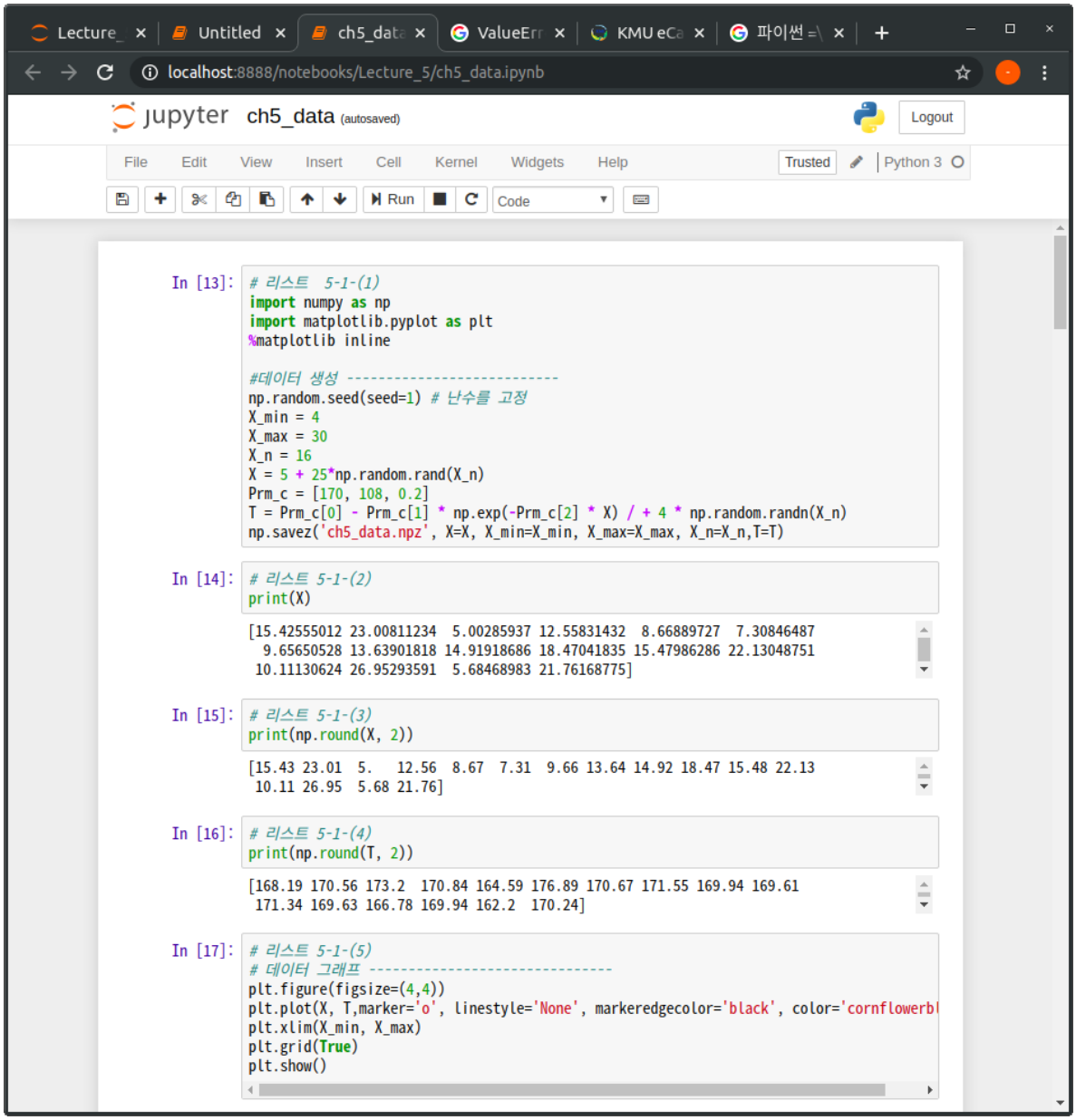
Homework #4

20171621 민대인

1. 목차

내용
1. 목차
2. 스크린샷
3. 간단한 소감


Lecture_
Untitled
ch5_data
ValueErr
KMU eCa
파이썬 =\
localhost:8888/notebooks/Lecture_5/ch5_data.ipynb
jupyter ch5_data (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [13]:
# 리스트 5-1-(1)
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
#데이터 생성 ----------
np.random.seed(seed=1) # 난수를 고정
X_min = 4
X_max = 30
X_n = 16
X = 5 + 25*np.random.rand(X_n)
Prm_c = [170, 108, 0.2]
T = Prm_c[0] - Prm_c[1] * np.exp(-Prm_c[2] * X) / + 4 * np.random.randn(X_n)
np.savez('ch5_data.npz', X=X, X_min=X_min, X_max=X_max, X_n=X_n,T=T)
In [14]:
# 리스트 5-1-(2)
print(X)
[15.42555012 23.00811234 5.00285937 12.55831432 8.66889727 7.30846487
9.65650528 13.63901818 14.91918686 18.47041835 15.47986286 22.13048751
10.11130624 26.95293591 5.68468983 21.76168775]
In [15]:
# 리스트 5-1-(3)
print(np.round(X, 2))
[15.43 23.01 5. 12.56 8.67 7.31 9.66 13.64 14.92 18.47 15.48 22.13
10.11 26.95 5.68 21.76]
In [16]:
# 리스트 5-1-(4)
print(np.round(T, 2))
[168.19 170.56 173.2 170.84 164.59 176.89 170.67 171.55 169.94 169.61
171.34 169.63 166.78 169.94 162.2 170.24]
In [17]:
# 리스트 5-1-(5)
# 데이터 그래프 ----------
plt.figure(figsize=(4,4))
plt.plot(X, T,marker='o', linestyle='None', markeredgecolor='black', color='cornflowerbl
plt.xlim(X_min, X_max)
plt.grid(True)
plt.show()

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

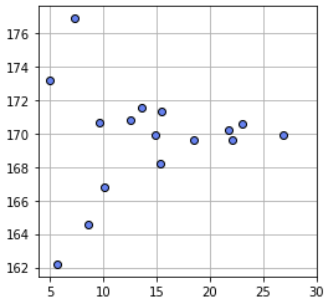
localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

jupyter ch5_data (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

```
plt.plot(X, T,marker='o', linestyle='None', markeredgecolor='black', color='cornflowerbl
plt.xlim(X_min, X_max)
plt.grid(True)
plt.show()
```



In [18]:

```
# 리스트 5-1-(6)
from mpl_toolkits.mplot3d import Axes3D
# 평균 오차 함수 --------------------------
def mse_line(x, t, w):
    y=w[0]*x+w[1]
    mse = np.mean((y-t)**2)
    return mse

# 계산 ----------------------------------
xn= 100 # 등고선 표시 해상도
w0_range=[-25, 25]
w1_range=[120,170]
x0=np.linspace(w0_range[0], w0_range[1], xn)
x1=np.linspace(w1_range[0], w1_range[1], xn)
xx0, xx1 = np.meshgrid(x0, x1)
J = np.zeros((len(x0), len(x1)))
for i0 in range(xn):
    for i1 in range(xn):
        J[i1, i0] = mse_line(X, T, (x0[i0], x1[i1]))

# 표시 ----------------------------------
plt.figure(figsize=(9.5,4))
plt.subplots_adjust(wspace=0.5)

ax=plt.subplot(1,2,1,projection='3d')
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

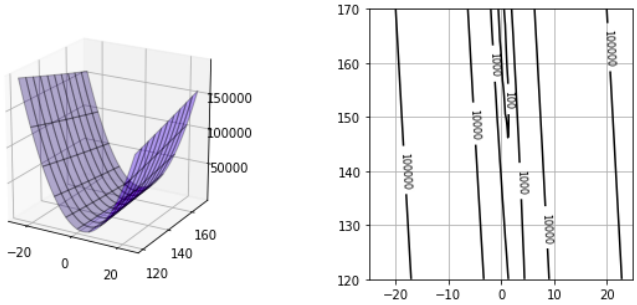
jupyter ch5_data (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

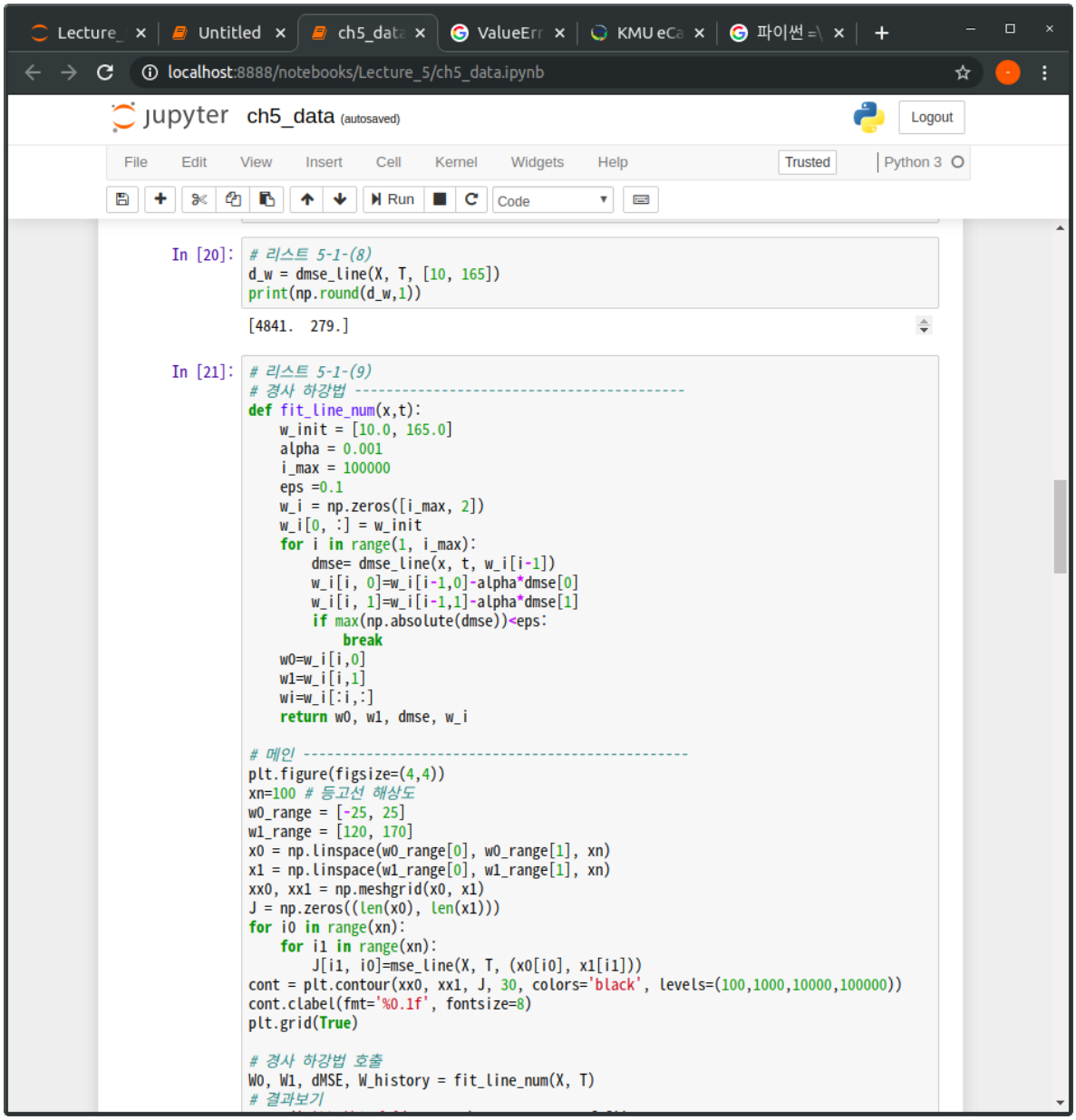
```
# 표시 ------------------------------------
plt.figure(figsize=(9.5,4))
plt.subplots_adjust(wspace=0.5)

ax=plt.subplot(1,2,1,projection='3d')
ax.plot_surface(xx0,xx1,J,rstride=10, cstride=10, alpha=0.3, color='blue', edgecolor='bl
ax.set_xticks([-20,0,20])
ax.set_yticks([120,140,160])
ax.view_init(20,-60)

plt.subplot(1,2,2)
cont=plt.contour(xx0,xx1, J ,30, colors='black', levels=[100,1000,10000,100000])
cont.clabel(fmt='%1.0f', fontsize=8)
plt.grid(True)
plt.show()
```



In [19]:
```
# 리스트 5-1-(7)
# 평균 제곱 오차의 기울기 ------------------------------
def dmse_line(x, t, w):
    y=w[0]*x+w[1]
    d_w0=2*np.mean((y-t)*x)
    d_w1=2*np.mean(y-t)
    return d_w0, d_w1
```

In [20]:
```
# 리스트 5-1-(8)
d_w = dmse_line(X, T, [10, 165])
print(np.round(d_w,1))
```

```
[4841.  279.]
```


Lecture_
Untitled
ch5_data
ValueErr
KMU eCa
파이썬 =\
localhost:8888/notebooks/Lecture_5/ch5_data.ipynb
jupyter ch5_data (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [20]:
# 리스트 5-1-(8)
d_w = dmse_line(X, T, [10, 165])
print(np.round(d_w,1))
[4841. 279.]
In [21]:
# 리스트 5-1-(9)
# 경사 하강법 ----------
def fit_line_num(x,t):
w_init = [10.0, 165.0]
alpha = 0.001
i_max = 100000
eps =0.1
w_i = np.zeros([i_max, 2])
w_i[0, :] = w_init
for i in range(1, i_max):
dmse= dmse_line(x, t, w_i[i-1])
w_i[i, 0]=w_i[i-1,0]-alpha*dmse[0]
w_i[i, 1]=w_i[i-1,1]-alpha*dmse[1]
if max(np.absolute(dmse))<eps:
break
w0=w_i[i,0]
w1=w_i[i,1]
wi=w_i[:i,:]
return w0, w1, dmse, w_i
# 메인 ----------
plt.figure(figsize=(4,4))
xn=100 # 등고선 해상도
w0_range = [-25, 25]
w1_range = [120, 170]
x0 = np.linspace(w0_range[0], w0_range[1], xn)
x1 = np.linspace(w1_range[0], w1_range[1], xn)
xx0, xx1 = np.meshgrid(x0, x1)
J = np.zeros((len(x0), len(x1)))
for i0 in range(xn):
for i1 in range(xn):
J[i1, i0]=mse_line(X, T, (x0[i0], x1[i1]))
cont = plt.contour(xx0, xx1, J, 30, colors='black', levels=(100,1000,10000,100000))
cont.clabel(fmt='%0.1f', fontsize=8)
plt.grid(True)
# 경사 하강법 호출
W0, W1, dMSE, W_history = fit_line_num(X, T)
결과보기

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

jupyter ch5_data (autosaved)

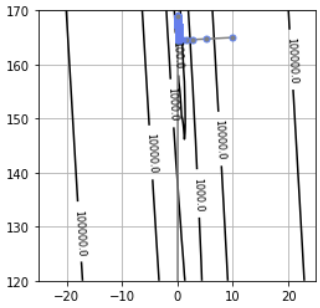
Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

```
cont.clabel(fmt='%0.1f', fontsize=8)
plt.grid(True)

# 경사 하강법 호출
W0, W1, dMSE, W_history = fit_line_num(X, T)
# 결과보기
print('반복 횟수 {0}'.format(W_history.shape[0]))
print('W=[{0:.6f}, {1:.6f}]'.format(dMSE[0],dMSE[1]))
print('MSE={0:.6f}'.format(mse_line(X,T,[W0,W1])))
plt.ylim(w1_range[0],w1_range[1])
plt.plot(W_history[:,0],W_history[:,1],'.-',color='gray', markersize=10, markeredgecolor
plt.show()
```

```
반복 횟수 100000
W=[0.005793, -0.099978]
MSE=10.343116
```



In [22]:

```
# 리스트 5-1-(10)
# 선 표시 ------------------------------------------------
def show_line(w):
    xb=np.linspace(X_min, X_max, 100)
    y = w[0]*xb + w[1]
    plt.plot(xb, y, color=(.5,.5,.5),linewidth=4)

# 메인 ---------------------------------------------------
plt.figure(figsize=(4,4))
W=np.array([W0,W1])
mse = mse_line(X,T,W)
print("w0={0:.3f}, w1={1:.3f}".format(W0, W1))
print("SD={0:.3f} cm".format(np.sqrt(mse)))
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | +

localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

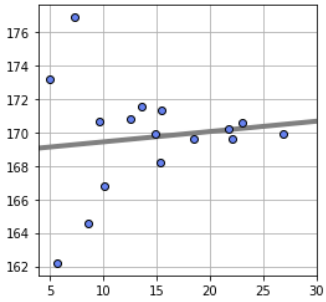
jupyter ch5_data (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

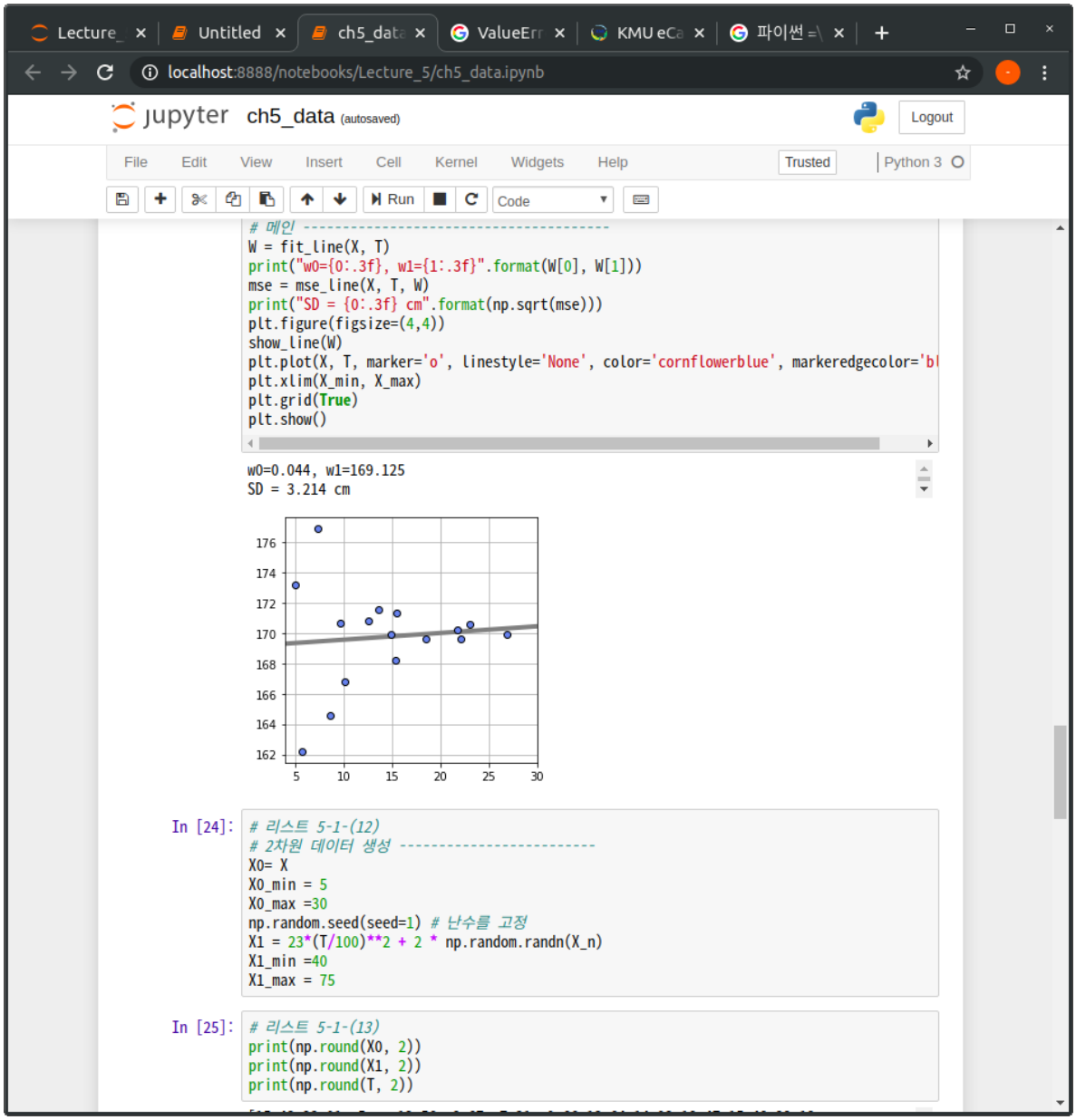
```
plt.figure(figsize=(4,4))
W=np.array([W0,W1])
mse = mse_line(X,T,W)
print("w0={0:.3f}, w1={1:.3f}".format(W0, W1))
print("SD={0:.3f} cm".format(np.sqrt(mse)))
show_line(W)
plt.plot(X, T, marker='o', linestyle='None', color='cornflowerblue', markeredgecolor='bl
plt.xlim(X_min, X_max)
plt.grid(True)
plt.show()
```

```
w0=0.062, w1=168.821
SD=3.216 cm
```



In [23]:

```
# 리스트 5-1-(11)
# 해석해 ------------------------------------
def fit_line(x, t):
    mx = np.mean(x)
    mt = np.mean(t)
    mtx = np.mean(t*x)
    mxx = np.mean(x*x)
    w0 = (mtx - mt * mx) / (mxx - mx**2)
    w1 = mt - w0*mx
    return np.array([w0, w1])
# 메인 ------------------------------------
W = fit_line(X, T)
print("w0={0:.3f}, w1={1:.3f}".format(W[0], W[1]))
mse = mse_line(X, T, W)
print("SD = {0:.3f} cm".format(np.sqrt(mse)))
plt.figure(figsize=(4,4))
```


Lecture_
Untitled
ch5_data
ValueErr
KMU eCa
파이썬 =\
localhost:8888/notebooks/Lecture_5/ch5_data.ipynb
jupyter ch5_data (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
# 메인 ----------
W = fit_line(X, T)
print("w0={0:.3f}, w1={1:.3f}".format(W[0], W[1]))
mse = mse_line(X, T, W)
print("SD = {0:.3f} cm".format(np.sqrt(mse)))
plt.figure(figsize=(4,4))
show_line(W)
plt.plot(X, T, marker='o', linestyle='None', color='cornflowerblue', markeredgecolor='b
plt.xlim(X_min, X_max)
plt.grid(True)
plt.show()
w0=0.044, w1=169.125
SD = 3.214 cm
176
174
172
170
168
166
164
162
5
10
15
20
25
30
In [24]:
# 리스트 5-1-(12)
# 2차원 데이터 생성 ----------
X0= X
X0_min = 5
X0_max =30
np.random.seed(seed=1) # 난수를 고정
X1 = 23*(T/100)**2 + 2 * np.random.randn(X_n)
X1_min =40
X1_max = 75
In [25]:
# 리스트 5-1-(13)
print(np.round(X0, 2))
print(np.round(X1, 2))
print(np.round(T, 2))

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

jupyter ch5_data (autosaved)

Logout

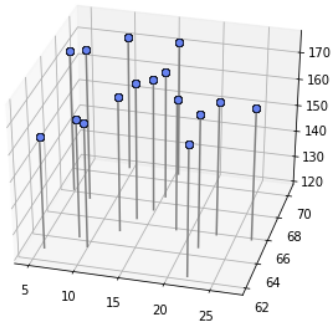
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

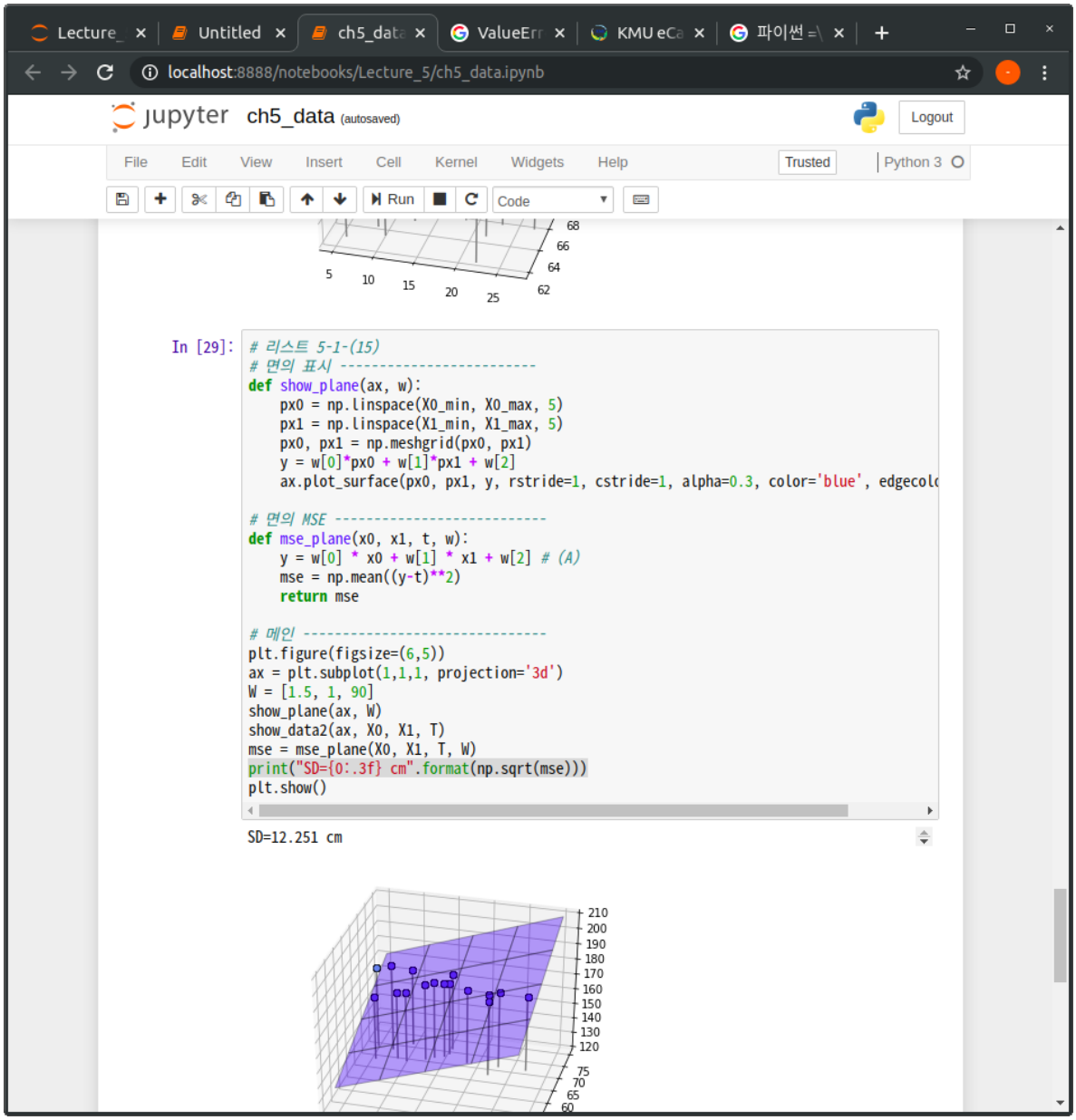
Run Code

In [25]:
```
# 리스트 5-1-(13)
print(np.round(X0, 2))
print(np.round(X1, 2))
print(np.round(T, 2))
```

```
[15.43 23.01  5.   12.56  8.67  7.31  9.66 13.64 14.92 18.47 15.48 22.13
 10.11 26.95  5.68 21.76]
[68.31 65.68 67.94 64.98 64.04 67.36 70.49 66.16 67.06 65.67 70.45 62.06
 63.33 65.65 62.78 64.46]
[168.19 170.56 173.2  170.84 164.59 176.89 170.67 171.55 169.94 169.61
 171.34 169.63 166.78 169.94 162.2  170.24]
```

In [26]:
```
# 리스트 5-1-(14)
# 2차원 데이터의 표시 ----------------------------
def show_data2(ax, x0, x1, t):
    for i in range(len(x0)):
        ax.plot([x0[i],x0[i]], [x1[i],x1[i]], [120, t[i]], color='gray')
        ax.plot(x0, x1, t, 'o', color='cornflowerblue', markeredgecolor='black', markers
        ax.view_init(elev=35, azim=-75)

# 메인 ----------------------------------------
plt.figure(figsize=(6,5))
ax = plt.subplot(1,1,1,projection='3d')
show_data2(ax, X0, X1, T)
plt.show()
```



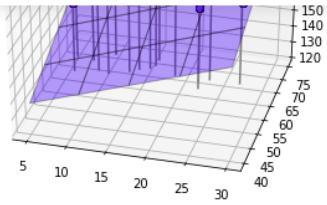

Lecture_
Untitled
ch5_data
ValueErr
KMU eCa
파이썬 =\
localhost:8888/notebooks/Lecture_5/ch5_data.ipynb
jupyter ch5_data (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [29]:
# 리스트 5-1-(15)
# 면의 표시 ------------------------
def show_plane(ax, w):
px0 = np.linspace(X0_min, X0_max, 5)
px1 = np.linspace(X1_min, X1_max, 5)
px0, px1 = np.meshgrid(px0, px1)
y = w[0]*px0 + w[1]*px1 + w[2]
ax.plot_surface(px0, px1, y, rstride=1, cstride=1, alpha=0.3, color='blue', edgecolo
# 면의 MSE ----------------------------
def mse_plane(x0, x1, t, w):
y = w[0] * x0 + w[1] * x1 + w[2] # (A)
mse = np.mean((y-t)**2)
return mse
# 메인 -------------------------------
plt.figure(figsize=(6,5))
ax = plt.subplot(1,1,1, projection='3d')
W = [1.5, 1, 90]
show_plane(ax, W)
show_data2(ax, X0, X1, T)
mse = mse_plane(X0, X1, T, W)
print("SD={0:.3f} cm".format(np.sqrt(mse)))
plt.show()
SD=12.251 cm

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

jupyter ch5_data (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

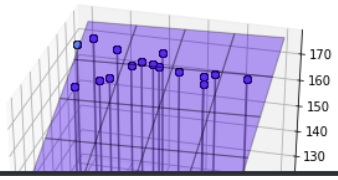
Run Code



In [31]:

```
# 리스트 5-1-(16)
# 해석해 ------------------------------------
def fit_plane(x0, x1, t):
    c_tx0 = np.mean(t*x0) - np.mean(t) * np.mean(x0)
    c_tx1 = np.mean(t*x1) - np.mean(t) * np.mean(x1)
    c_x0x1 = np.mean(x0 * x1) - np.mean(x0) * np.mean(x1)
    v_x0 = np.var(x0)
    v_x1 = np.var(x1)
    w0 = (c_tx1 * c_x0x1 - v_x1 * c_tx0) / (c_x0x1**2 - v_x0*v_x1)
    w1 = (c_tx0 * c_x0x1 - v_x0 * c_tx1) / (c_x0x1**2 - v_x0*v_x1)
    w2 = -w0 * np.mean(x0) - w1 * np.mean(x1) + np.mean(t)
    return np.array([w0,w1,w2])

# 메인 ----------------------------------------
plt.figure(figsize=(6,5))
ax = plt.subplot(1,1,1,projection='3d')
W = fit_plane(X0, X1, T)
print("w0={0:.3f}, w1={1:.3f}, w2={2:.3f}".format(W[0], W[1], W[2]))
show_plane(ax, W)
show_data2(ax, X0, X1, T)
mse = mse_plane(X0, X1, T, W)
print("SD={0:.3f} cm".format(np.sqrt(mse)))
plt.show()
```

```
w0=0.088, w1=0.741, w2=119.531
SD=2.688 cm
```



Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/ch5_data.ipynb

jupyter ch5_data (autosaved)

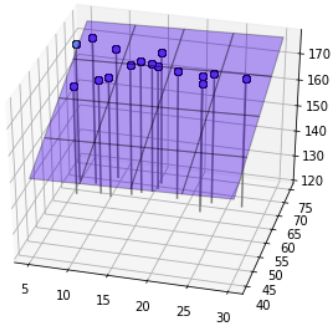
Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

```
    c_tx1 = np.mean(t*x1) - np.mean(t) * np.mean(x1)
    c_x0x1 = np.mean(x0 * x1) - np.mean(x0) * np.mean(x1)
    v_x0 = np.var(x0)
    v_x1 = np.var(x1)
    w0 = (c_tx1 * c_x0x1 - v_x1 * c_tx0) / (c_x0x1**2 - v_x0*v_x1)
    w1 = (c_tx0 * c_x0x1 - v_x0 * c_tx1) / (c_x0x1**2 - v_x0*v_x1)
    w2 = -w0 * np.mean(x0) - w1 * np.mean(x1) + np.mean(t)
    return np.array([w0,w1,w2])

# 메인 ------------------------------------
plt.figure(figsize=(6,5))
ax = plt.subplot(1,1,1,projection='3d')
W = fit_plane(X0, X1, T)
print("w0={0:.3f}, w1={1:.3f}, w2={2:.3f}".format(W[0], W[1], W[2]))
show_plane(ax, W)
show_data2(ax, X0, X1, T)
mse = mse_plane(X0, X1, T, W)
print("SD={0:.3f} cm".format(np.sqrt(mse)))
plt.show()
```

```
w0=0.088, w1=0.741, w2=119.531
SD=2.688 cm
```



Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter **Untitled** (autosaved) Logout

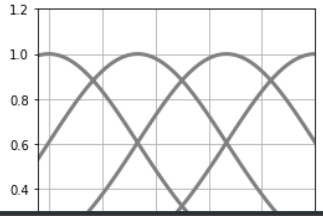
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [1]:
```
# --- 리스트 5-2-(1)
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

# 데이터 로드 -------------------------
outfile = np.load('ch5_data.npz')
X = outfile['X']
X_min =outfile['X_min']
X_max = outfile['X_max']
X_n = outfile['X_n']
T = outfile['T']
```

In [5]:
```
# --- 리스트 5-2-(2)
# 가우스 함수 ---------------------------
def gauss(x, mu, s):
    return np.exp(-(x-mu)**2 /(2*s**2))
```

In [6]:
```
# 리스트 5-2-(3)
# 메인 ----------------------------------
M = 4
plt.figure(figsize=(4,4))
mu = np.linspace(5,30, M)
s = mu[1] -mu[0] # (A)
xb = np.linspace(X_min, X_max, 100)
for j in range(M):
    y =gauss(xb, mu[j], s)
    plt.plot(xb, y, color='gray', linewidth=3)
plt.grid(True)
plt.xlim(X_min, X_max)
plt.ylim(0, 1.2)
plt.show()
```

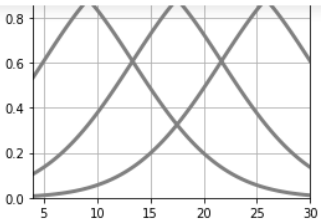


Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter Untitled (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code



In [7]:
```
# 리스트 5-2-(4)
# 선형 기저 함수 모델 ----------------------
def gauss_func(w, x):
    m = len(w) -1
    mu = np.linspace(5, 30, m)
    s = mu[1] -mu[0]
    y = np.zeros_like(x) # x와 같은 크기로 요소가 0의 행렬 y를 작성
    for j in range(m):
        y = y + w[j] * gauss(x, mu[j], s)
    y = y + w[m]
    return y
```

In [77]:
```
# 리스트 5-2-(5)
# 선현 기저 함수 모델 MSE --------------------
def mse_gauss_func(x, t, w):
    y = gauss_func(w, x)
    mse =np.mean((y - t)**2)
    return mse
```

In [78]:
```
# 리스트 5-2-(6)
# 선형 기저 함수 모델 정확한 해 ---------------
def fit_gauss_func(x, t, m):
    mu = np.linspace(5,30,m)
    s= mu[1]- mu[0]
    n = x.shape[0]
    psi = np.ones((n, m+1))
    for j in range(m):
        psi[:, j] = gauss(x, mu[j], s)
    psi_T = np.transpose(psi)

    b = np.linalg.inv(psi_T.dot(psi))
    c = b.dot(psi_T)
    w = c.dot(t)
    return w
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter **Untitled** (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

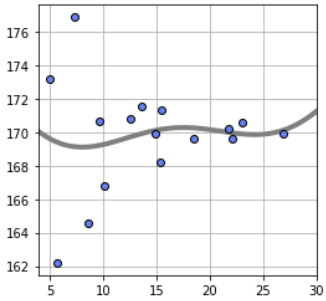
Run Code

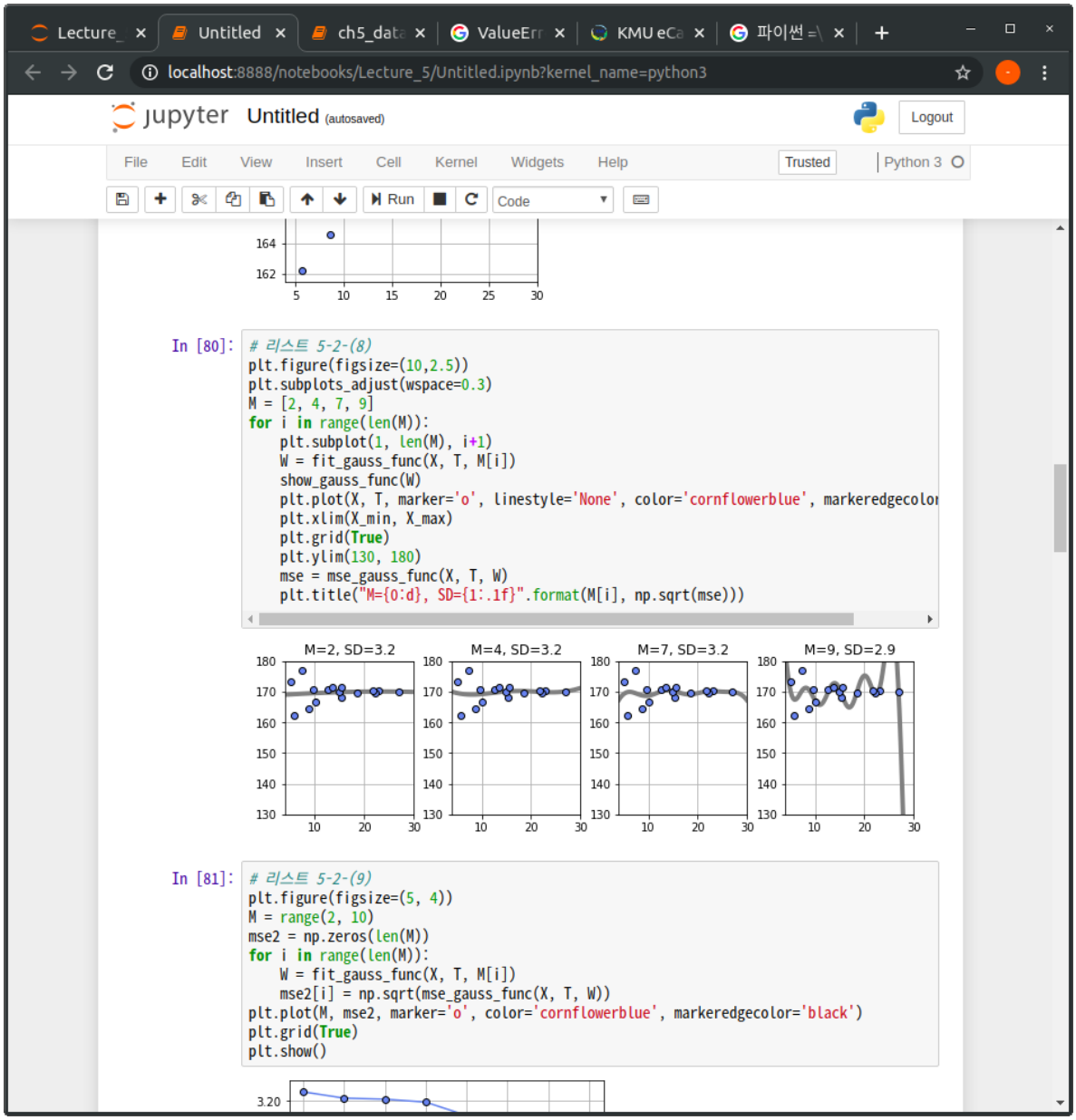
```
    psi_T = np.transpose(psi)

    b = np.linalg.inv(psi_T.dot(psi))
    c = b.dot(psi_T)
    w = c.dot(t)
    return w
```

In [79]:

```
# 리스트 5-2-(7)
# 가우스 기저 함수 표시 ----------------------
def show_gauss_func(w):
    xb = np.linspace(X_min, X_max, 100)
    y = gauss_func(w, xb)
    plt.plot(xb, y, c=[.5,.5,.5], lw=4)

# 메인 ---------------------------------------
plt.figure(figsize=(4,4))
M = 4
W = fit_gauss_func(X,T,M)
show_gauss_func(W)
plt.plot(X, T, marker='o', linestyle='None', color='cornflowerblue', markeredgecolor='bl
plt.xlim(X_min, X_max)
plt.grid(True)
mse = mse_gauss_func(X, T, W)
print('W='+str(np.round(W,1)))
print("SD = {0:.2f} cm".format(np.sqrt(mse)))
plt.show()
```

```
W=[-11.8  -1.9  -5.6  -8.4 183.4]
SD = 3.20 cm
```




Lecture_
Untitled
ch5_data
ValueErr
KMU eCa
파이썬 =\
localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3
jupyter Untitled (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [80]:
# 리스트 5-2-(8)
plt.figure(figsize=(10,2.5))
plt.subplots_adjust(wspace=0.3)
M = [2, 4, 7, 9]
for i in range(len(M)):
plt.subplot(1, len(M), i+1)
W = fit_gauss_func(X, T, M[i])
show_gauss_func(W)
plt.plot(X, T, marker='o', linestyle='None', color='cornflowerblue', markeredgecolor
plt.xlim(X_min, X_max)
plt.grid(True)
plt.ylim(130, 180)
mse = mse_gauss_func(X, T, W)
plt.title("M={0:d}, SD={1:.1f}".format(M[i], np.sqrt(mse)))
M=2, SD=3.2
M=4, SD=3.2
M=7, SD=3.2
M=9, SD=2.9
In [81]:
# 리스트 5-2-(9)
plt.figure(figsize=(5, 4))
M = range(2, 10)
mse2 = np.zeros(len(M))
for i in range(len(M)):
W = fit_gauss_func(X, T, M[i])
mse2[i] = np.sqrt(mse_gauss_func(X, T, W))
plt.plot(M, mse2, marker='o', color='cornflowerblue', markeredgecolor='black')
plt.grid(True)
plt.show()

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

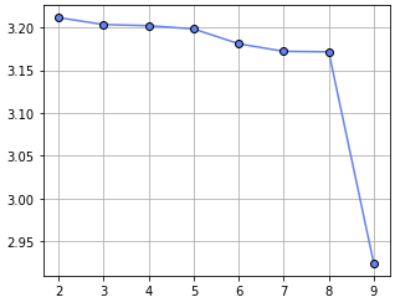
localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter Untitled (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

```
plt.plot(M, mse2, marker='o', color='cornflowerblue', markeredgecolor='black')
plt.grid(True)
plt.show()
```



In [82]:

```
# 리스트 5-2-(10)
# 훈련 데이터와 테스트 데이터 ---------------------
X_test = X[:int(X_n / 4 + 1)]
T_test = T[:int(X_n / 4 + 1)]
X_train = X[int(X_n / 4 + 1):]
T_train = T[int(X_n / 4 + 1):]
# 메인 ---------------------------------------
plt.figure(figsize=(10, 2.5))


plt.subplots_adjust(wspace=0.3)
M = [2, 4, 7, 9]
for i in range(len(M)):
    plt.subplot(1, len(M), i + 1)
    W = fit_gauss_func(X_train, T_train, M[i])
    show_gauss_func(W)
    plt.plot(X_train, T_train, marker='o', color='white', markeredgecolor='black', label
    plt.plot(X_test, T_test, marker='o', color='cornflowerblue', markeredgecolor='black'
    plt.legend(loc='lower right', fontsize=10, numpoints=1)
    plt.xlim(X_min, X_max)
    plt.ylim(130, 180)
    plt.grid(True)
    mse = mse_gauss_func(X_test, T_test, W)
    plt.title("M={0:d}, SD={1:.1f}".format(M[i], np.sqrt(mse)))
plt.show()
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | +

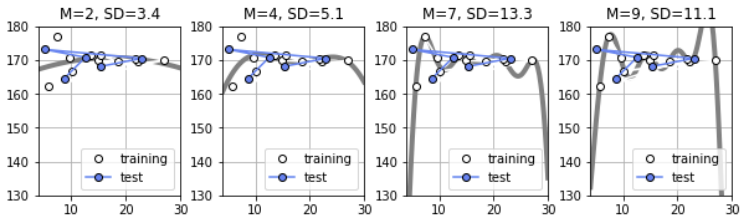
localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter Untitled (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

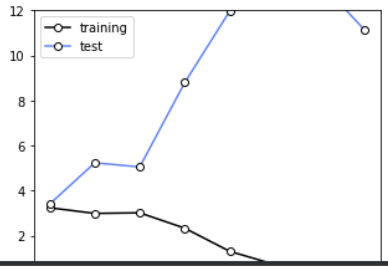
```
    plt.grid(True)
    mse = mse_gauss_func(X_test, T_test, W)
    plt.title("M={0:d}, SD={1:.1f}".format(M[i], np.sqrt(mse)))
plt.show()
```



In [83]:

```
# 리스트 5-2-(11)
plt.figure(figsize=(5, 4))
M = range(2, 10)
mse_train = np.zeros(len(M))
mse_test = np.zeros(len(M))
for i in range(len(M)):
    W = fit_gauss_func(X_train, T_train, M[i])
    mse_train[i] = np.sqrt(mse_gauss_func(X_train, T_train, W))
    mse_test[i] = np.sqrt(mse_gauss_func(X_test, T_test, W))
plt.plot(M, mse_train, marker='o', markerfacecolor='white', markeredgecolor='black', col
plt.plot(M, mse_test, marker='o', markerfacecolor='white', markeredgecolor='black', colo
plt.legend(loc='upper left', fontsize=10)
plt.ylim(0, 12)
```

Out[83]: (0.0, 12.0)

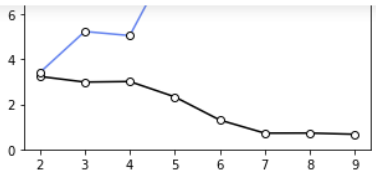


Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter Untitled (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code



In [87]:
```
# 리스트 5-2-(12)
# K겹 교차 검증 ---------------------------------------
def kfold_gauss_func(x, t, m, k):
    n = x.shape[0]
    mse_train = np.zeros(k)
    mse_test = np.zeros(k)
    for i in range(0, k):
        x_train = x[np.fmod(range(n), k) != i] # (A)
        t_train = t[np.fmod(range(n), k) != i] # (A)
        x_test = x[np.fmod(range(n), k) == i] # (A)
        t_test = t[np.fmod(range(n), k) == i] # (A)
        wm = fit_gauss_func(x_train, t_train, m)

        mse_train[i] = mse_gauss_func(x_train, t_train, wm)
        mse_test[i] = mse_gauss_func(x_test, t_test, wm)
    return mse_train, mse_test
```

In [88]:
```
# 리스트 5-2-(13)
np.fmod(range(10), 5)
```

Out[88]:
```
array([0, 1, 2, 3, 4, 0, 1, 2, 3, 4])
```

In [89]:
```
# 리스트 5-2-(14)
M = 4
K = 4
kfold_gauss_func(X, T, M, K)
```

Out[89]:
```
(array([ 9.25633247,  4.87774416,  5.57408757, 13.14858355]),
 array([  23.55403547, 1308.00873405,  120.19314282,    2.56056657]))
```

In [108]:
```
# 리스트 5-2-(15)
M = range(2, 8)
K = 16
Cv_Gauss_train = np.zeros((K, len(M)))
Cv_Gauss_test = np.zeros((K, len(M)))
for i in range(0, len(M)):
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | +

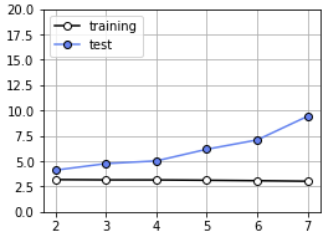
localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter Untitled (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [108]:
```
# 리스트 5-2-(15)
M = range(2, 8)
K = 16
Cv_Gauss_train = np.zeros((K, len(M)))
Cv_Gauss_test = np.zeros((K, len(M)))
for i in range(0, len(M)):
    Cv_Gauss_train[:, i], Cv_Gauss_test[:, i] = kfold_gauss_func(X, T, M[i], K)
    mean_Gauss_train = np.sqrt(np.mean(Cv_Gauss_train, axis = 0))
    mean_Gauss_test = np.sqrt(np.mean(Cv_Gauss_test, axis = 0))


plt.figure(figsize=(4, 3))
plt.plot(M, mean_Gauss_train, marker='o', color='k', markerfacecolor='w', label='trainir
plt.plot(M, mean_Gauss_test, marker='o', color='cornflowerblue', markeredgecolor='black'
plt.legend(loc='upper left', fontsize=10)
plt.ylim(0, 20)
plt.grid(True)
plt.show()
```



In [109]:
```
# 리스트 5-2-(16)
M = 3
plt.figure(figsize=(4,4))
W = fit_gauss_func(X, T, M)
show_gauss_func(W)
plt.plot(X, T, marker='o', linestyle='None', color='cornflowerblue', markeredgecolor='bl
plt.xlim([X_min, X_max])
plt.grid(True)
mse = mse_gauss_func(X, T, W)
print("SD={0:.2f} cm".format(np.sqrt(mse)))
plt.show()
```

```
SD=3.20 cm
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\ | +

localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

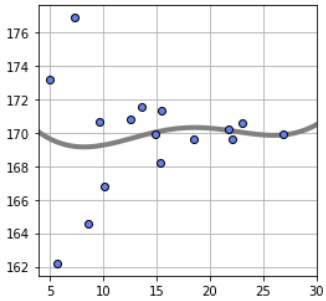
jupyter Untitled (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

```
print("SD={0:.2f} cm".format(np.sqrt(mse)))
plt.show()
```

```
SD=3.20 cm
```



In [111]:

```
# 리스트 5-2-(17)
# 모델 A ---------------------------------
def model_A(x, w):
    y = w[0] - w[1] * np.exp(-w[2] * x)
    return y

# 모델 A 표시 ----------------------------
def show_model_A(w):
    xb = np.linspace(X_min, X_max, 100)
    y = model_A(xb, w)
    plt.plot(xb, y, c=[.5, .5, .5], lw = 4)

# 모델 A의 MSE ---------------------------
def mse_model_A(w, x, t):
    y = model_A(x, w)
    mse = np.mean((y - t)**2)
    return mse
```

In [112]:

```
# 리스트 5-2-(18)
from scipy.optimize import minimize

# 모델 A의 매개 변수 최적화 ---------------
def fit_model_A(w_init, x, t):
    res1 = minimize(mse_model_A, w_init, args=(x, t), method="powell")
```

Lecture_ | Untitled | ch5_data | ValueErr | KMU eCa | 파이썬 =\

localhost:8888/notebooks/Lecture_5/Untitled.ipynb?kernel_name=python3

jupyter Untitled (autosaved) Logout

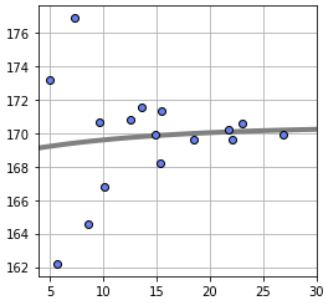
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

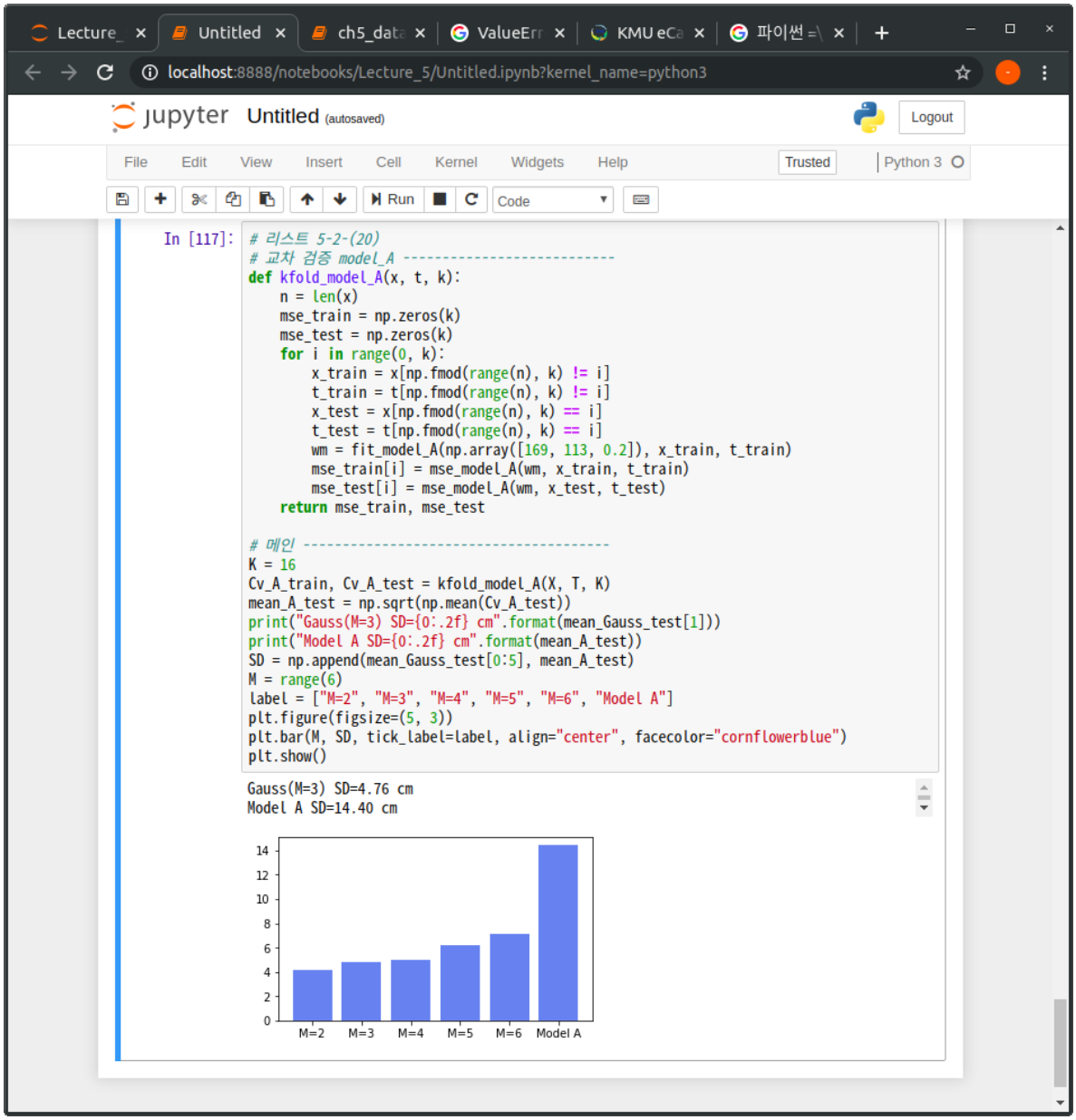
In [112]:
```
# 리스트 5-2-(18)
from scipy.optimize import minimize

# 모델 A의 매개 변수 최적화 ---------------
def fit_model_A(w_init, x, t):
    res1 = minimize(mse_model_A, w_init, args=(x, t), method="powell")
    return res1.x
```

In [114]:
```
# 리스트 5-2-(19)
# 메인 ---------------------------------
plt.figure(figsize=(4, 4))
W_init=[100,0,0]
W = fit_model_A(W_init, X, T)
print("w0={0:.1f}, w1={1:.1f}, w2={2:.1f}".format(W[0], W[1], W[2]))
show_model_A(W)
plt.plot(X, T, marker='o', linestyle='None', color='cornflowerblue', markeredgecolor='bl
plt.xlim(X_min, X_max)
plt.grid(True)
mse = mse_model_A(W, X, T)
print("SD={0:.2f} cm".format(np.sqrt(mse)))
plt.show()
```

```
w0=170.4, w1=1.8, w2=0.1
SD=3.21 cm
```



In [117]:
```
# 리스트 5-2-(20)
# 교차 검증 model_A ---------------------------
def kfold_model_A(x, t, k):
    n = len(x)
    mse_train = np.zeros(k)
```



---

3. 간단한 소감

그래프를 그려보면 데이터 시각화를 재밌게 공부할 수 있었다.