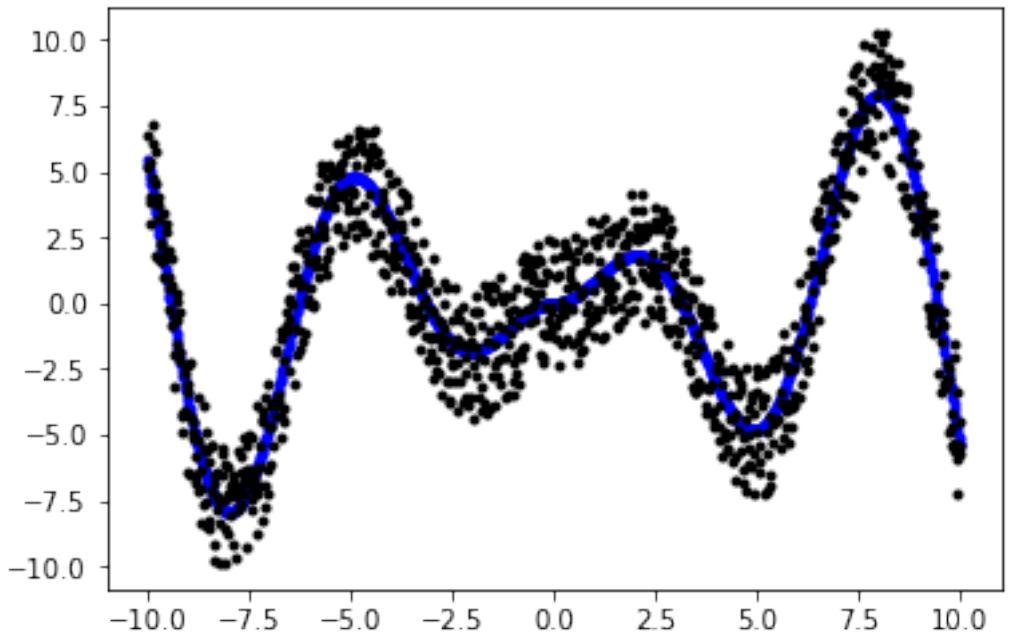
Assignment 12

December 20, 2018

name : Jaehong Park
student ID : 2017120175
Github Link : https://github.com/westsun0920/Assignment12

In [29]:
```
import numpy as np
import matplotlib.pyplot as plt

num     = 1001
std     = 5

# x  : x-coordinate data
# y1 : (clean) y-coordinate data
# y2 : (noisy) y-coordinate data

def fun(x):

        # f = np.sin(x) * (1 / (1 + np.exp(-x)))
        f = np.abs(x) * np.sin(x)

        return f

n       = np.random.rand(num)
nn      = n - np.mean(n)
x       = np.linspace(-10,10,num)
y1      = fun(x)                        # clean points
y2      = y1 + nn * std                 # noisy points

plt.plot(x, y1, 'b.', x, y2, 'k.')
plt.show()
```




```
In [30]: def leastSquare(v1,v2):
             v1 = np.mat(v1)
             v2 = np.mat(v2).T
             v3 = v1.T * v1
             theta = v3.I * (v1.T * v2) # weight
             return theta

In [31]: def computeLambda(matrix, vec, lambda_):
             column = len(matrix.T)
             lamvec = np.ones((1,column), dtype=float)
             zerovec = [0]
             matrixA = np.concatenate((matrix, lambda_*lamvec), axis=0)
             matrixY = np.concatenate((vec, zerovec), axis=0)
             return matrixA, matrixY

In [32]: def computefunction(n , lam):
             dimensions = []; dimenX = []; var = []; weight = []; y = []; y_ = []
             for i in range(0, len(x)):
                 dimenX = []
                 for j in range(0, n+1):
                     dimenX.append(x[i]**j)
                 dimensions.append(dimenX)  # Set X0 = 1.0 for computing
             dimensions = np.mat(dimensions)
             matrixA, matrixY = computeLambda(dimensions, y2, lam)
             weight = leastSquare(matrixA, matrixY)
             for j in range(0, n+1):
```