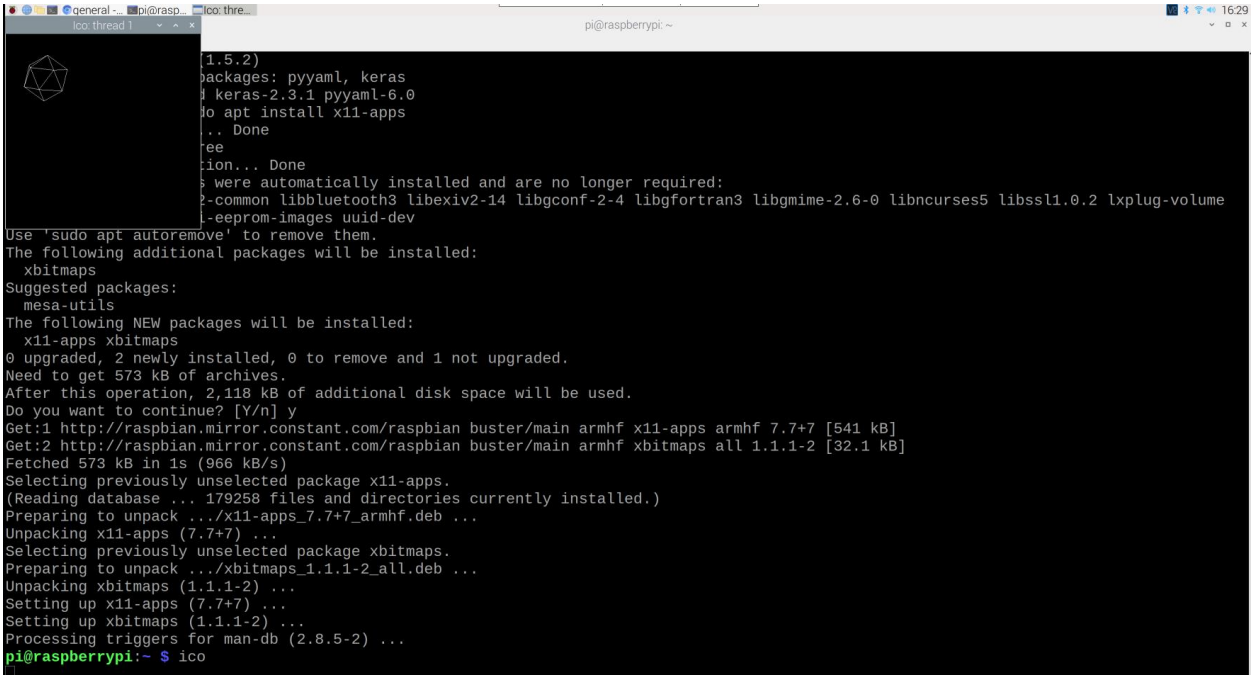
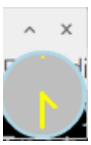
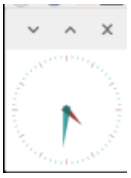
Lab 8 — Data Analysis

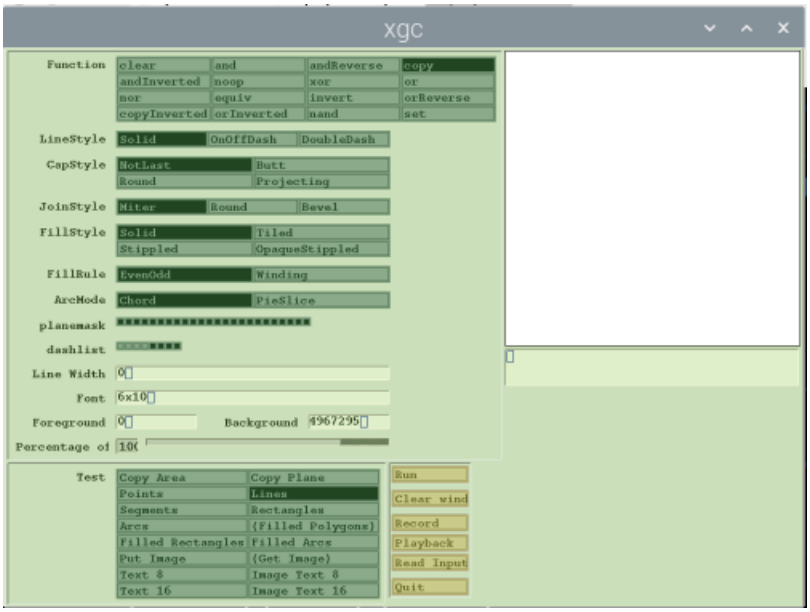
- Study the GitHub repository Lesson 8
- Install Python packages
- Save the Lab 7 Google sheet in CSV format to ~/demo
- Copy ~/iot/lesson8/plt_final.py and plt_cv2.py to ~/demo
- Edit plt_final.py and plt_cv2.py to read the CSV file
- Run plt_final.py and plt_cv2.py





Calculator
0
1/x
x²
√‾
CE/C
AC
INV
sin
cos
tan
DRG
e
EE
log
ln
y^x
π
x!
(
)
÷
STO
7
8
9
*
RCL
4
5
6
-
SUM
1
2
3
+
EXC
0
.
+/-
=

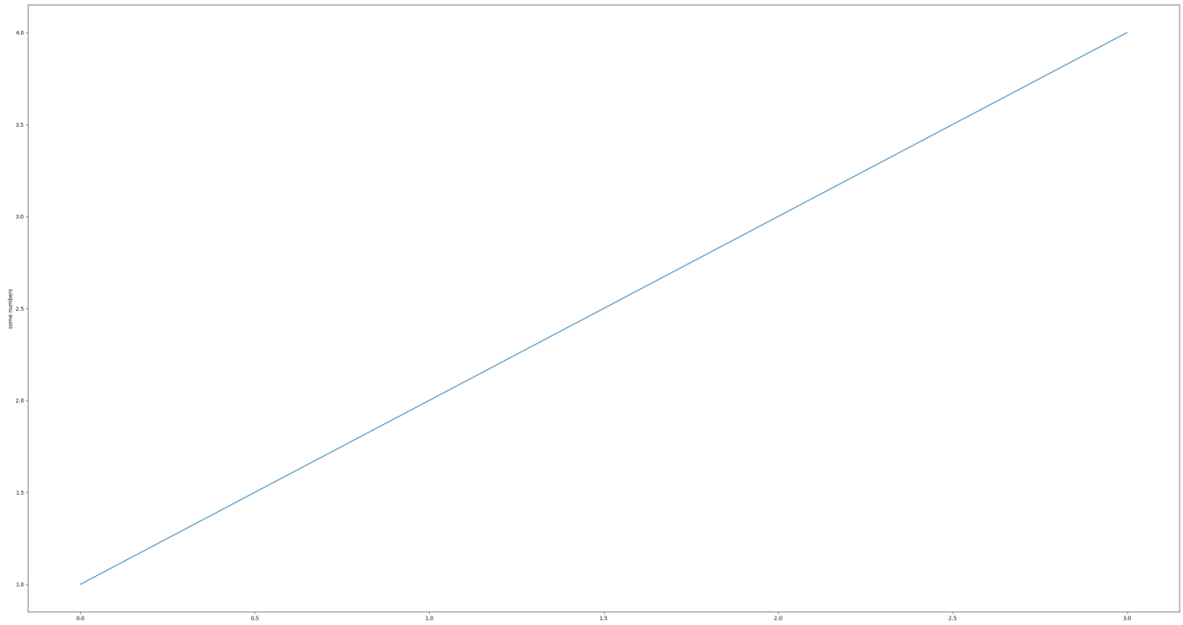


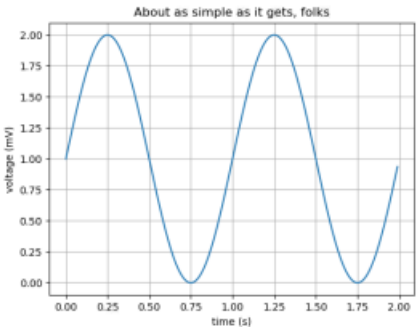

xgc
Function
clear
and
andReverse
copy
andInverted
noop
xor
or
nor
equiv
invert
orReverse
copyInverted
orInverted
nand
set
LineStyle
Solid
OnOffDash
DoubleDash
CapStyle
NotLast
Butt
Round
Projecting
JoinStyle
Miter
Round
Bevel
FillStyle
Solid
Tiled
Stippled
OpaqueStippled
FillRule
EvenOdd
Winding
ArcMode
Chord
PieSlice
planemask
dashlist
Line Width
0
Font
6x10
Foreground
0
Background
4967295
Percentage of
Test
Copy Area
Copy Plane
Points
Lines
Segments
Rectangles
Arcs
(Filled Polygons)
Filled Rectangles
Filled Arcs
Put Image
(Get Image)
Text 8
Image Text 8
Text 16
Image Text 16
Run
Clear wind
Record
Playback
Read Input
Quit

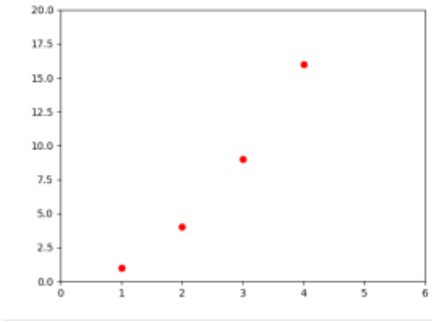
```
$ python3
>>> import numpy as np
>>> a = np.arange(6)
>>> a
>>> b = np.arange(12).reshape(4, 3)
>>> b
>>> c = np.arange(24).reshape(2, 3, 4)
>>> c
>>> b.shape
>>> b.reshape(-1)
>>> b.reshape(-1, 1)
>>> b.reshape(2, -1)
>>> d = np.array([20, 30, 40, 50])
>>> e = np.arange(4)
>>> f = d-e
>>> f
>>> e**2
>>> A = np.array([[1, 1], [0, 1]])
>>> B = np.array([[2, 0], [3, 4]])
>>> A*B
>>> A.dot(B)
>>> np.dot(A, B)
>>> g = np.ones((2, 3), dtype=int)
>>> g
>>> h = np.random.random((2, 3))
>>> h
>>> g *= 3
>>> g
>>> h += g
>>> h
>>> k = np.random.random((2, 3))
>>> k
>>> k.sum()
>>> k.min()
>>> k.max()
>>> m = np.arange(12).reshape(3, 4)
>>> m
>>> m.sum(axis = 0)
>>> m.min(axis = 1)
>>> m.cumsum(axis = 1)
>>> n = np.arange(5)
>>> n
>>> n[[1, 3, 4]] = 0
>>> n
>>> exit()
```

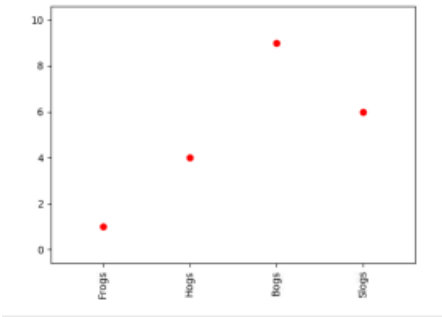
```
>>> g
array([[3, 3, 3],
       [3, 3, 3]])
>>> h += g
>>> h
array([[3.62645876, 3.91983062, 3.27322645],
       [3.25033093, 3.0129131 , 3.21140803]])
>>> k = np.random.random((2, 3))
>>> k
array([[0.233694  , 0.00550868, 0.2250866 ],
       [0.61305934, 0.37214455, 0.72000954]])
>>> k.sum()
2.1695027217435006
>>> k.min()
0.005508683430435091
>>> k.max()
0.7200095425642735
>>> m = np.arange(12).reshape(3, 4)
>>> m
array([[ 0,  1,  2,  3],
       [ 4,  5,  6,  7],
       [ 8,  9, 10, 11]])
>>> m.sum(axis = 0)
array([12, 15, 18, 21])
>>> m.min(axis = 1)
array([0, 4, 8])
>>> m.cumsum(axis = 1)
array([[ 0,  1,  3,  6],
       [ 4,  9, 15, 22],
       [ 8, 17, 27, 38]], dtype=int32)
>>> n = np.arange(5)
>>> n
array([0, 1, 2, 3, 4])
>>> n[[1, 3, 4]] = 0
>>> n
array([0, 0, 2, 0, 0])
```

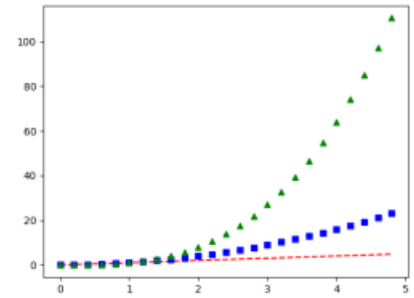
```
$ cd ~/iot/lesson8
$ python3 pyplot_simple.py
$ python3 simple_plot.py
$ python3 pyplot_formatstr.py
$ python3 ticklabels_demo_rotation.py
$ python3 pyplot_three.py
$ python3 pyplot_two_subplots.py
$ python3 pyplot_scales.py
$ python3 pyplot_annotate.py
$ python3 major_minor_demo1.py
$ python3 legend_demo.py
```

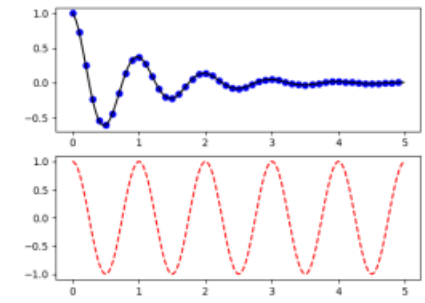


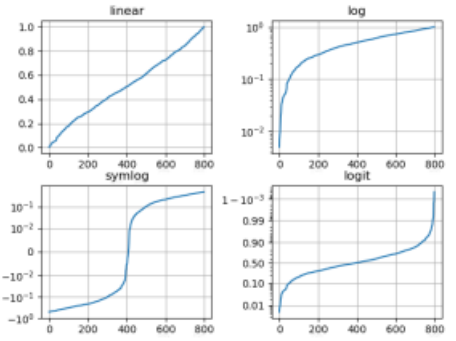


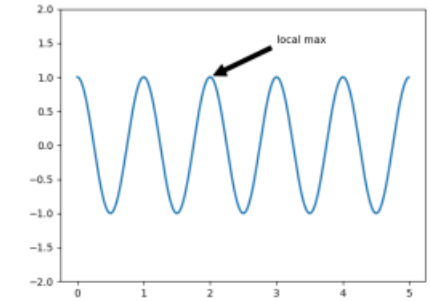


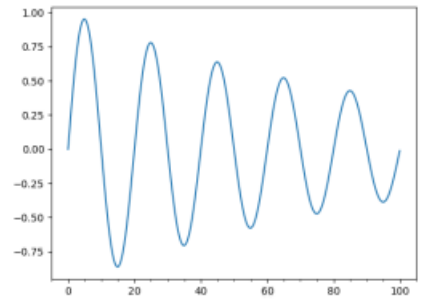

Frogs
Hogs
Bogs
Slogs

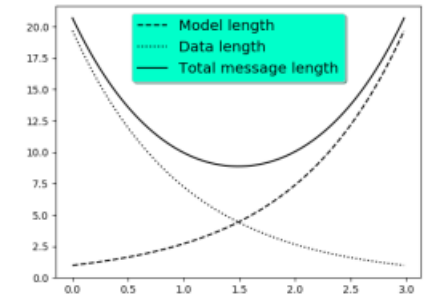


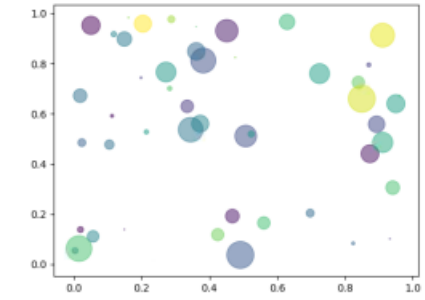


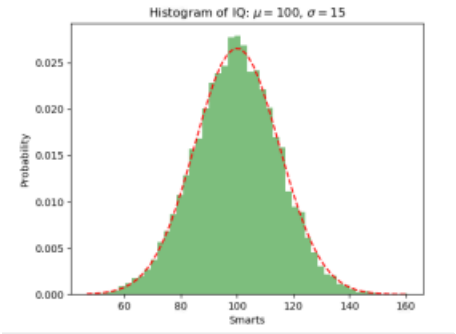

linear
log
symlog
logit

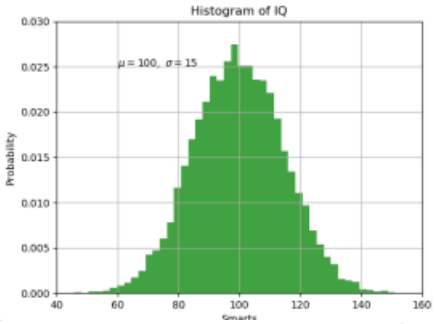

local max

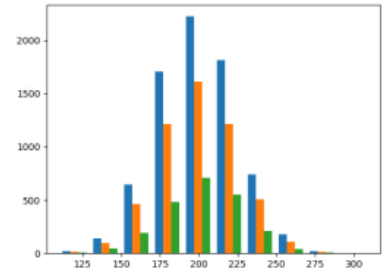


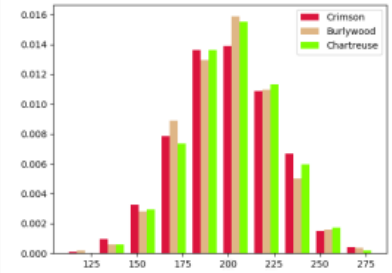

Model length
Data length
Total message length











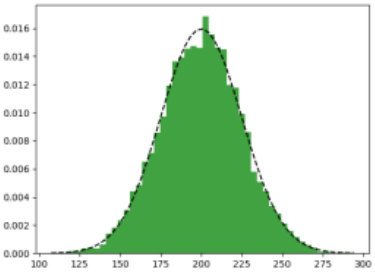
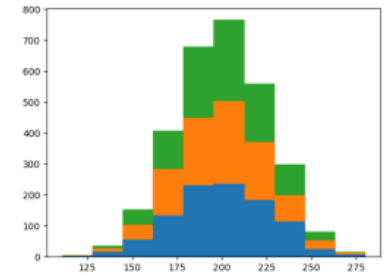


Figure 6

Figure 3



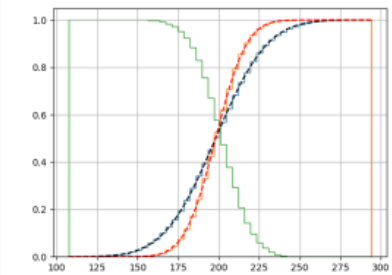
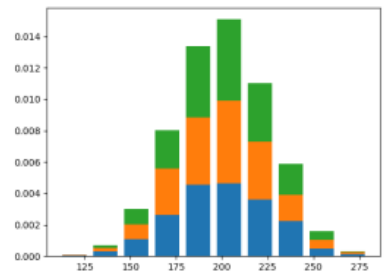
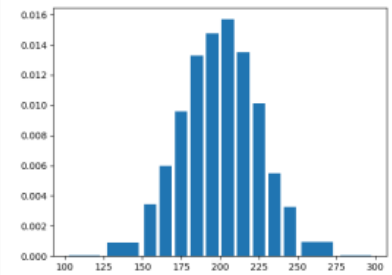


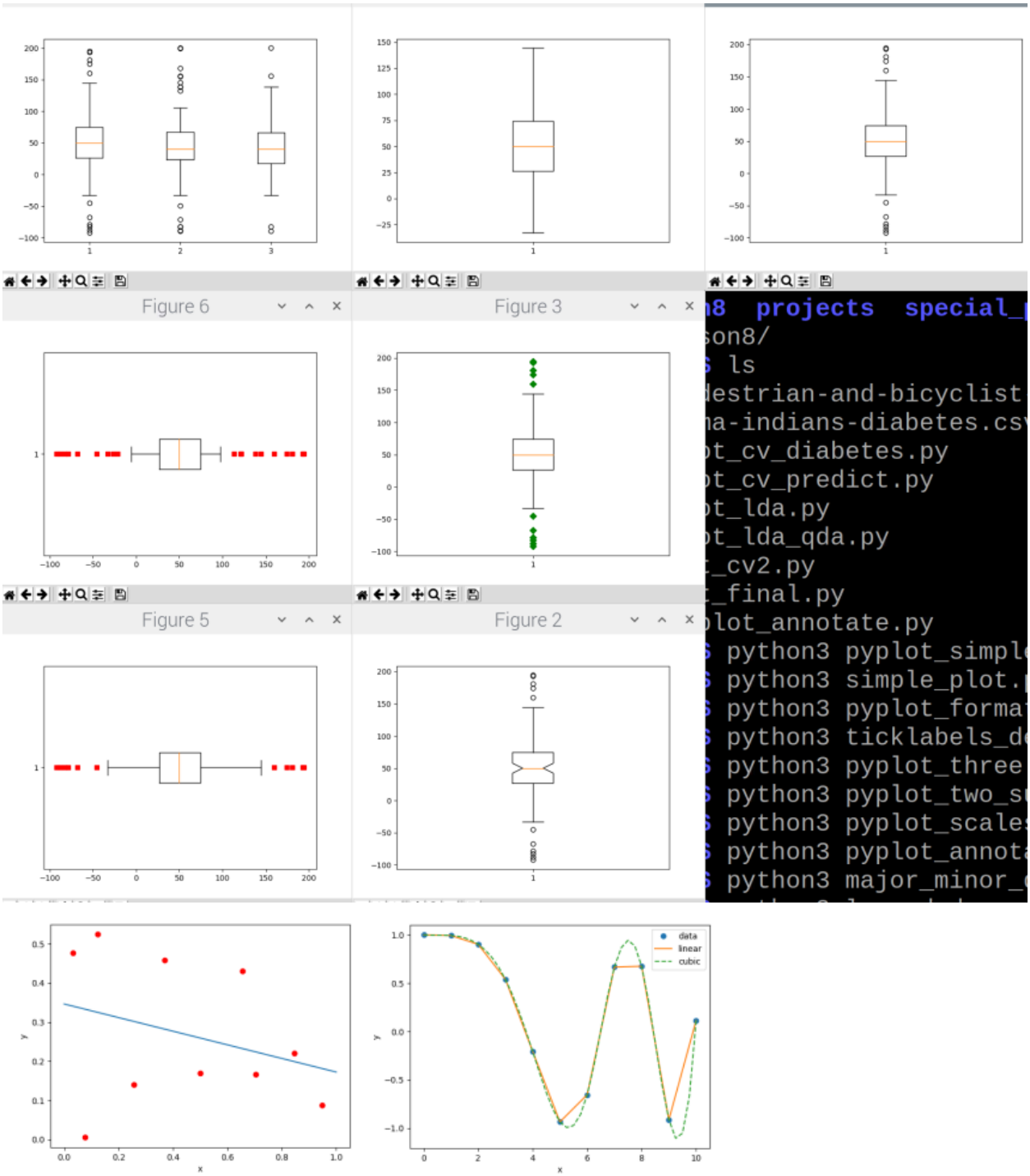
Figure 5

Figure 2





```
8  projects  special_
son8/
$ ls
destrian-and-bicyclist
a-indians-diabetes.csv
t_cv_diabetes.py
t_cv_predict.py
t_lda.py
t_lda_qda.py
_cv2.py
_final.py
lot_annotate.py
$ python3 pyplot_simple
$ python3 simple_plot.p
$ python3 pyplot_forma
$ python3 ticklabels_d
$ python3 pyplot_three
$ python3 pyplot_two_s
$ python3 pyplot_scale
$ python3 pyplot_annot
$ python3 major_minor_
```



Will update with Lab8 part b (4/29/22)