```
#Allows the importing of packages
using Pkg

#Adds necessary packages
Pkg.add("DataFrames")
Pkg.add("CSV")
Pkg.add("PyPlot")
Pkg.add("Plots")
Pkg.add("PyCall")
Pkg.add("SciPy")

#Load in dataframe and csv packages
using DataFrames
using PyCall
using CSV
using PyPlot
using SciPy

#load data in using Dataframe casting
#Need to preserve comments when importing the
data =
DataFrame(CSV.File("Kcomplexity_log_redund_4x1_10gts5lb_5funcs.csv"))

#Create standard plot
#Plots.scatter(data.kcomp, data.log_redund)

#Importing python numpy module for vstack function
np = pyimport("numpy")

#Creating heat density plot with matlab plotting method
x = data.kcomp
y = data.log_redund

#Calculating point density

#Stacks the vectors vertically
xy = np.vstack([x, y])
```

```
z = SciPy.stats.gaussian_kde(xy)(xy)

#Customizing the plot
fig, ax = PyPlot.subplots()
PyPlot.scatter(x, y, c = z, s=10, edgecolor="none", cmap = "Reds")


PyPlot.tick_params(labelsize=15)
PyPlot.xticks(size = 15, family = "Times New Roman") #make text fit to the
png
PyPlot.yticks(size = 15, family = "Times New Roman")
PyPlot.xlabel("Kolmogorov complexity", size = 20, family = "Times New
Roman")
PyPlot.ylabel("log redundancy", size = 20, family = "Times New Roman")
cb = PyPlot.colorbar(shrink = .5)
cb.ax.tick_params(labelsize = 15)
for i in cb.ax.yaxis.get_ticklabels()
    i.set_family("Times New Roman")
end

PyPlot.figtext(0.76, 0.73, "density", size = 20, family = "Times New
Roman")

PyPlot.savefig("C:\\Users\\Owner\\julia_RA\\heat_density\\rudementaryplotj
uliaWOKDE_8_23_22.png")
print("Hello")
```

The following packages were used:

DataFrames, CSV, PyPlot, PyCall, SciPy

DataFrames and CSV for data importing purposes. PyPlot for plotting functionality, specifically matplotlib utilization, Pycall to use pyimport in order to use the python module numpy in Julia and SciPy to use kernel density estimation.

The data is loaded into a dataframe using the following.
`data=DataFrame(CSV.File("Kcomplexity_log_redund_4x1_10gts5lb_5funcs.csv"))`

The function pyimport is used to import the python module numpy. The variable np is shorthand for the module.

```
np = pyimport("numpy")
```

Desired variables are determined and assigned to x and y.

```
x = data.kcomp
y = data.log_redund
```

Using the function vstack, the vectors x and y are “stacked” on one another.

```
xy = np.vstack([x, y])
```

A gaussian kernel density estimation is used in order to give weights to data points. This allows for the color gradient seen in the heat map.

```
z = SciPy.stats.gaussian_kde(xy)(xy)
```

A subplot is created, assigning the resulting duple to fig and ax.

```
fig, ax = PyPlot.subplots()
```

Creates a scatter plot using desired x and y variables, with the color weighting being derived from the gaussian_kde, a size of 10, no edge color, and a color map of red.

```
PyPlot.scatter(x, y, c = z, s=10, edgecolor="none", cmap = "Reds")
```

Changes the label size, and the font, in addition to labels for the axes.

```
PyPlot.tick_params(labelsize=15)
PyPlot.xticks(size = 15, family = "Times New Roman")
PyPlot.yticks(size = 15, family = "Times New Roman")
PyPlot.xlabel("Kolmogorov complexity", size = 20, family = "Times New
Roman")
PyPlot.ylabel("log redundancy", size = 20, family = "Times New Roman")
```

Sizing and font

```
cb = PyPlot.colorbar(shrink = .5)
cb.ax.tick_params(labelsize = 15)
for i in cb.ax.yaxis.get_ticklabels()
    i.set_family("Times New Roman")
end
```

```
PyPlot.figtext(0.76, 0.73, "density", size = 20, family = "Times New
Roman")

PyPlot.savefig("..")
```