Topic No. 10

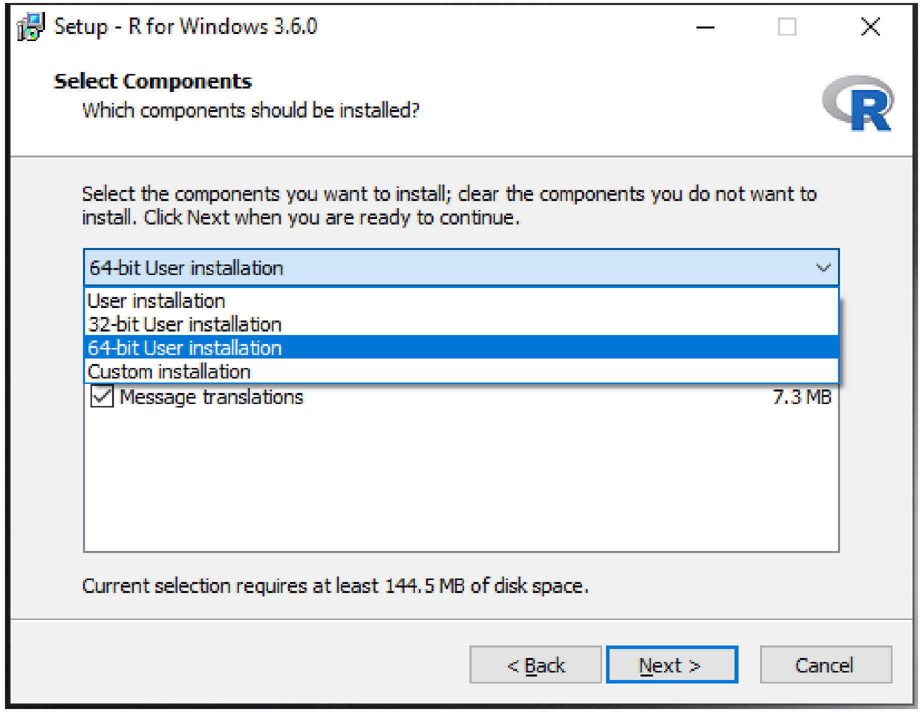
1. ggplot2 - R
2. 2D vs. 3D
3. Kaplan-Meier Survival analysis – R
4. KM in Python DASH

How to install R on Windows PC (Mac should be similar)

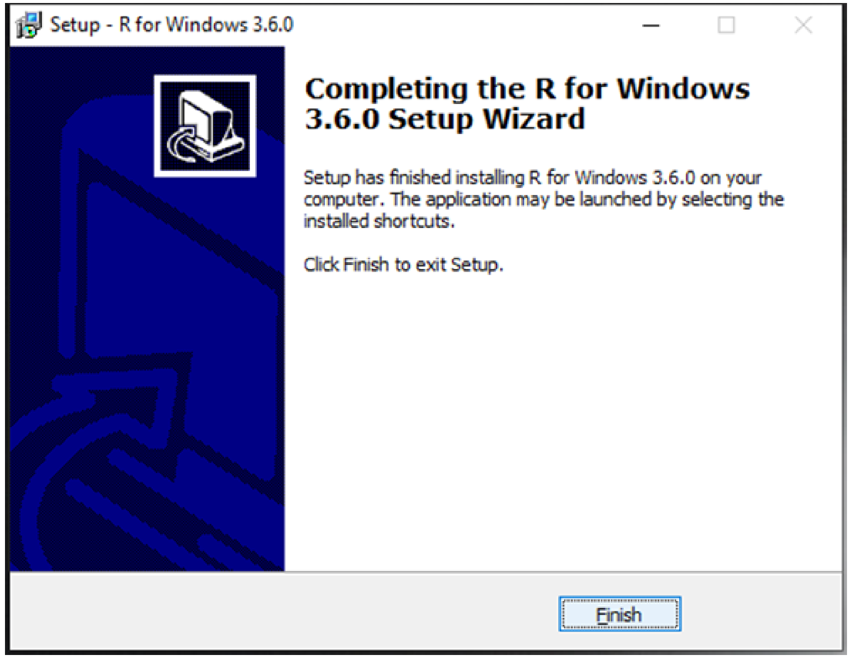
System requirement: Windows 10 (32/64 bit)

Step 1 Set up R environment:

1. Download R install package from http://lib.stat.cmu.edu/R/CRAN/bin/windows/base/old/3.6.0/R-3.6.0-win.exe
2. Run the set up program.



3. Change the installation according to your OS at this step.
4. Keep click "Next" button until the installation process is completed.

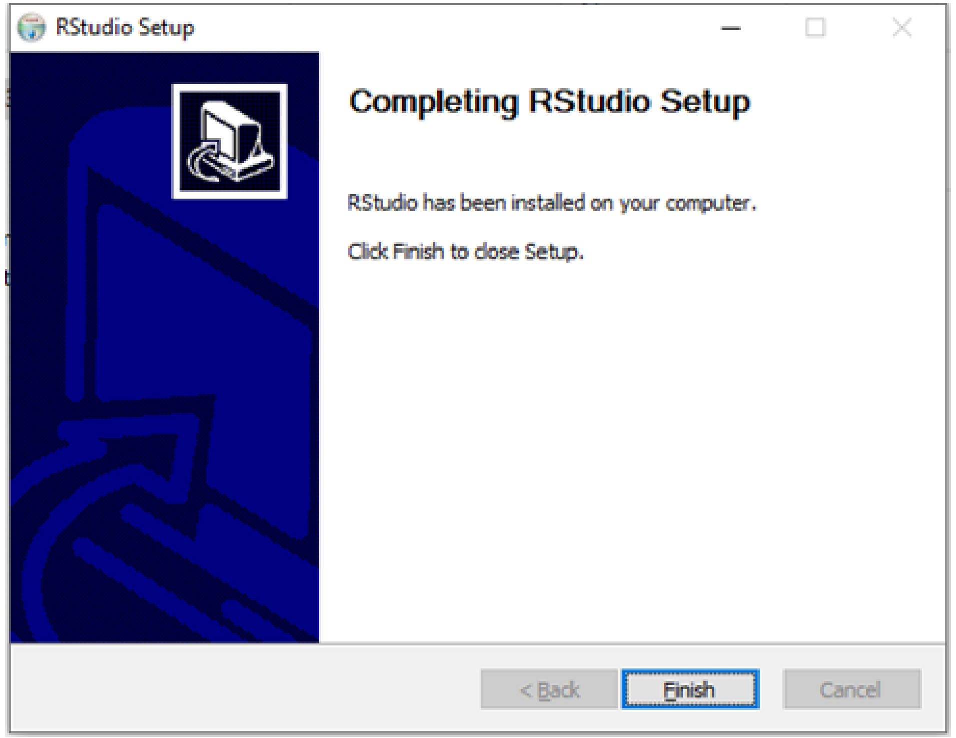


Step 2 Install IDE of Rstudio (optional):

1. Download Rstudio package from

   https://download1.rstudio.org/desktop/windows/RStudio-1.4.1103.exe

1. Run the set up program.
2. Keep click “Next” button until the installation process is completed.



Install.packages('ggplot2')

library(ggplot2)

Functions in ggplot2 (3.3.5)

https://www.rdocumentation.org/packages/ggplot2/versions/3.3.5

https://www.r-graph-gallery.com/ggplot2-package.html

Matplotlib VS Ggplot2

The Python vs R visualization showdown we have all been waiting for. - Rebecca Patro

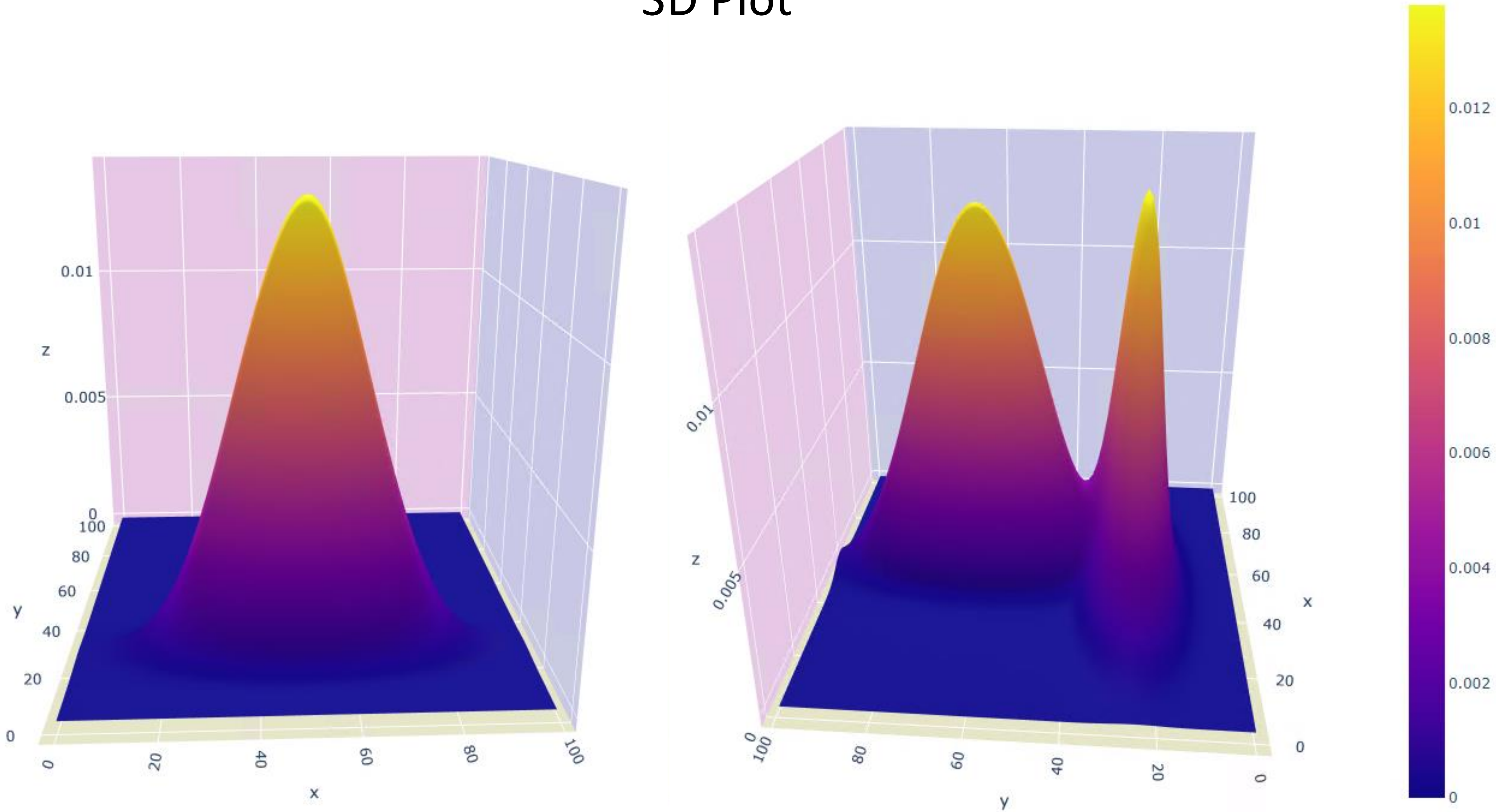
Ggplot2 (R) wins this visualization battle!

Both the packages are powerful tools for visualization. In the hands of a more skilled practitioner than me, they can yield better results. Matplotlib can create beautiful graphs and has a polished presentation style. **The reason ggplot2 won out was in its data handling capabilities**. If I allowed myself to use other packages as well python could have won.

	Matplotlib (python)	Ggplot2 (R)
Round 1: Scatter Plot	5	5
Round 2: Contour Plot	5	4
Round 3: Heatmap	5	5
Round 4: Regression multiline	4	5
Round 5: Multiline connected	5	4
Round 6 Polar chart	3	5
Round 7: Multiple Boxplots	3	5
Round 8: Bonus	5	4
Total	**35**	**37**

https://towardsdatascience.com/matplotlib-vs-ggplot2-c86dd35a9378

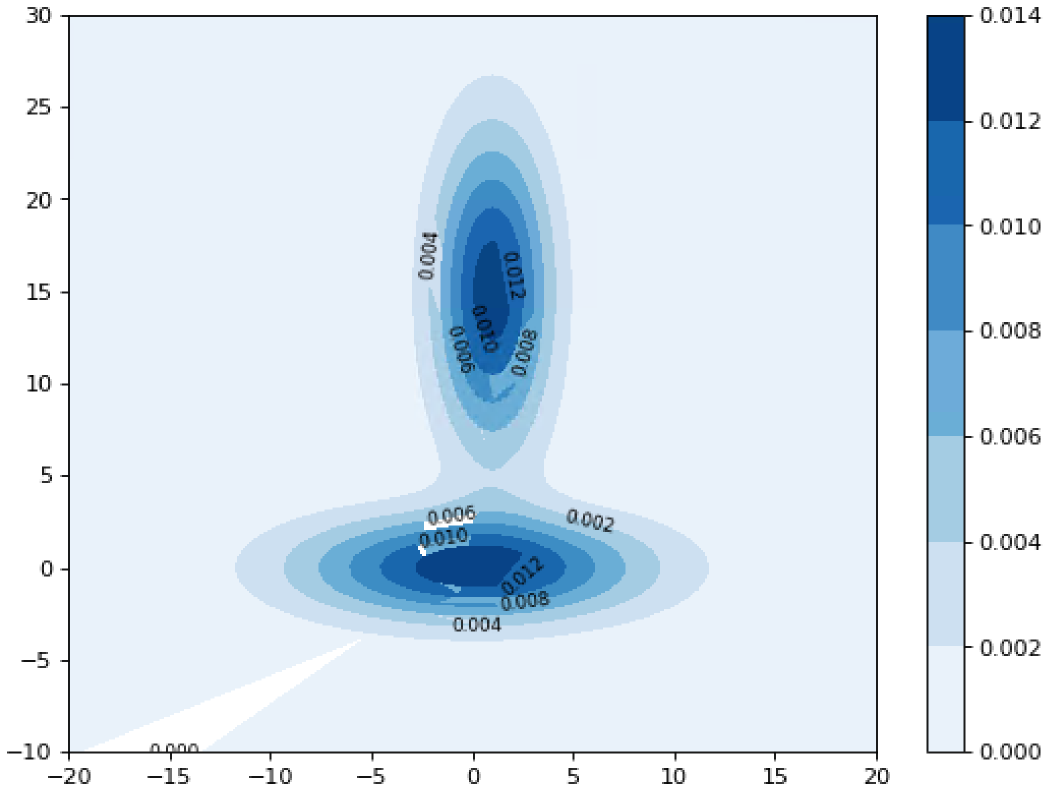
3D Plot



0.01
z
0.005
0
100
80
60
y
40
20
0
0
20
40
60
80
100
x
0.01
z
0.005
100
80
60
x
40
20
0
0
100
80
60
40
20
0
y
0.012
0.01
0.008
0.006
0.004
0.002
0

```
import plotly.graph_objects as go

fig = go.Figure(data=[go.Surface(z=z_data1+z_data2)])

fig.update_layout(title='3D Plot', autosize=False, width=700, height=700, margin=dict(l=65, r=50, b=65, t=90))

fig.update_layout(scene = dict(
                xaxis = dict(
                     backgroundcolor="rgb(200, 200, 230)",
                     gridcolor="white", showbackground=True,
                     zerolinecolor="white",),
                yaxis = dict(
                    backgroundcolor="rgb(230, 200,230)",
                    gridcolor="white", showbackground=True,
                    zerolinecolor="white"),
                zaxis = dict(
                    backgroundcolor="rgb(230, 230,200)",
                    gridcolor="white", showbackground=True,
                    zerolinecolor="white",),),
                width=700,
                margin=dict(
                r=10, l=10,
                b=10, t=10)
              )

fig.show()
```

2D Plot



http://localhost:8888/lab/tree/2D_3D.ipynb

```
import numpy as np
from matplotlib import pyplot as plt
from matplotlib.pyplot import figure

size = 100
sigma_x1 = 6.
sigma_y1 = 2.
sigma_x2 = 2.
sigma_y2 = 6.
m_x2= 1.
m_y2=15.

x = np.linspace(-20, 20, size)
y = np.linspace(-10, 30, size)

x, y = np.meshgrid(x, y)
z_data1 = (1/(2*np.pi*sigma_x1*sigma_y1) * np.exp(-(x**2/(2*sigma_x1**2)
    + y**2/(2*sigma_y1**2))))
z_data2 = (1/(2*np.pi*sigma_x2*sigma_y2) * np.exp(-((x-
m_x2)**2/(2*sigma_x2**2)
    + (y-m_y2)**2/(2*sigma_y2**2))))

figure(figsize=(8, 6), dpi=80)
contours=plt.contourf(x, y, z_data1+z_data2, cmap='Blues')
plt.clabel(contours, inline=True, fontsize=8, colors='black')
plt.colorbar()
plt.show()
```

Kaplan-Meier Survival Analysis

1. Want to know probability of survival (success) rate of a certain group of objects in terms of time.
   1) fraction of patients living for a certain amount of time after treatment
   2) length of time people remain unemployed after a job loss
   3) time-to-failure of machine parts
   4) how long fleshy fruits remain on plants before they are removed by animals
2. x-axis is for time, y-axis is for survival rate
3. two pieces of data are required for each patient (or each subject)
   1) the status at last observation (event occurrence or right-censored)
   2) the time to event (or time to censoring)

https://en.wikipedia.org/wiki/Kaplan%E2%80%93Meier_estimator

Kaplan-Meier Survival Analysis

1. *T* denotes the positive random variable representing time to event of interest.
2. Cumulative Distribution function : F(t) = Pr(T ≤ t)
3. Survival function :

$$S(t) = P(T > t | T > t-1) = P(T > t-1)$$

$$\widehat{S}(t) = \prod_{i:\ t_i \le t} \left(1 - \widehat{h}_i\right) = \prod_{i:\ t_i \le t} \left(1 - \frac{d_i}{n_i}\right)$$

d_i : Number of events at time t_i

n_i : Total individuals at risk at time t_i

Kaplan-Meier Survival Analysis

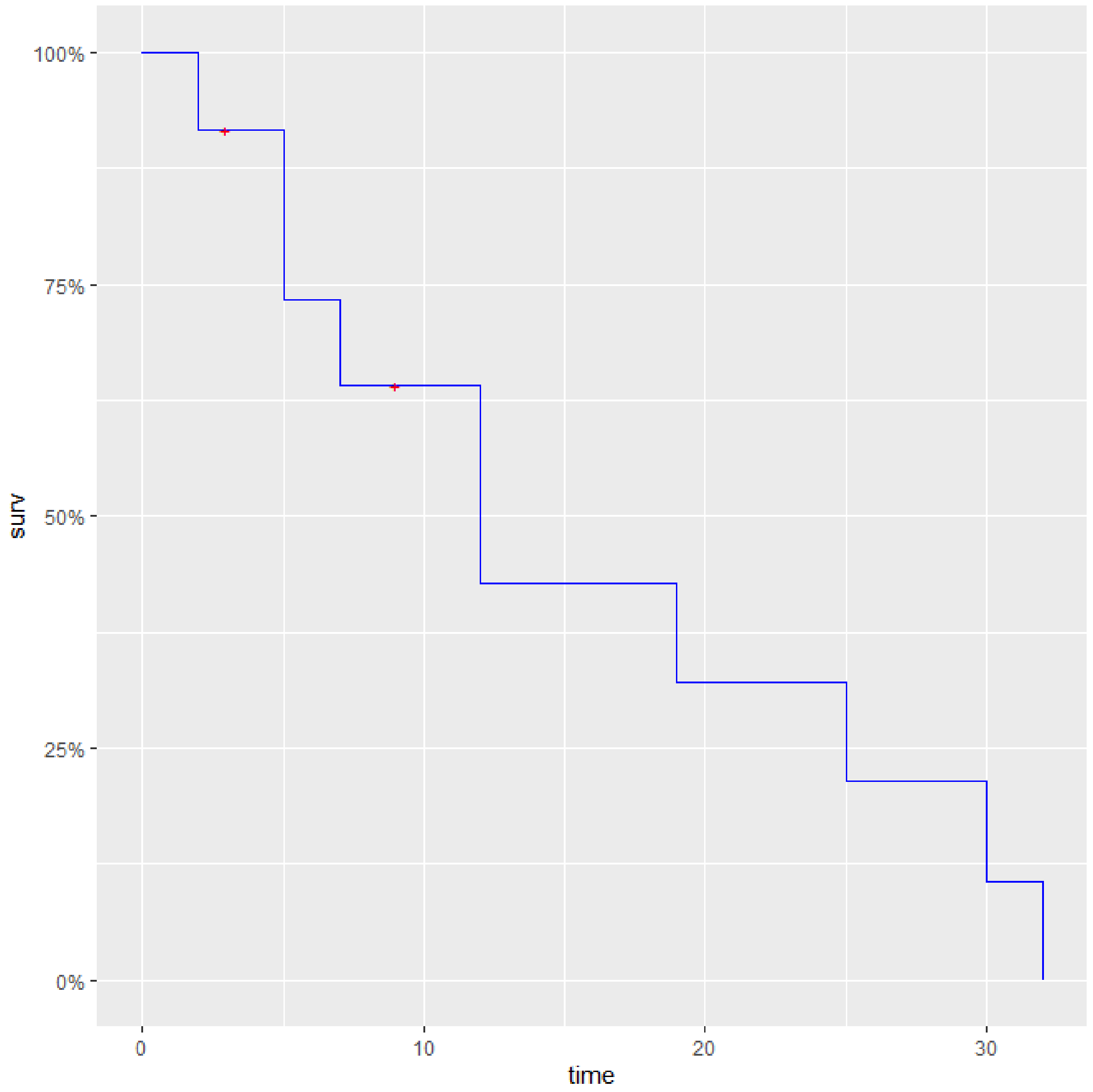
time	status	# of risk	# of dead	h_i	$1-h_i$	$S(t)$	$S(t)$
0		12	0	0/12	12/12	1	1
2	1	12	1	1/12	11/12	$S(0)$ *11/12	0.917
3	0						
5	1						
5	1	10	2	2/10	8/10	$S(2)$ *8/10	0.733
7	1	8	1	1/8	7/8	$S(5)$ *7/8	0.642
9	0						
12	1						
12	1	6	2	2/6	4/6	$S(7)$ *4/6	0.428
19	1	4	1	1/4	3/4	$S(12)$ *3/4	0.321
25	1	3	1	1/3	2/3	$S(19)$ *2/3	0.214
30	1	2	1	1/2	1/2	$S(25)$ *1/2	0.107
32	1	1	1	1/1	0/1	$S(30)$ *0/1	0

1 : dead
0 : censored

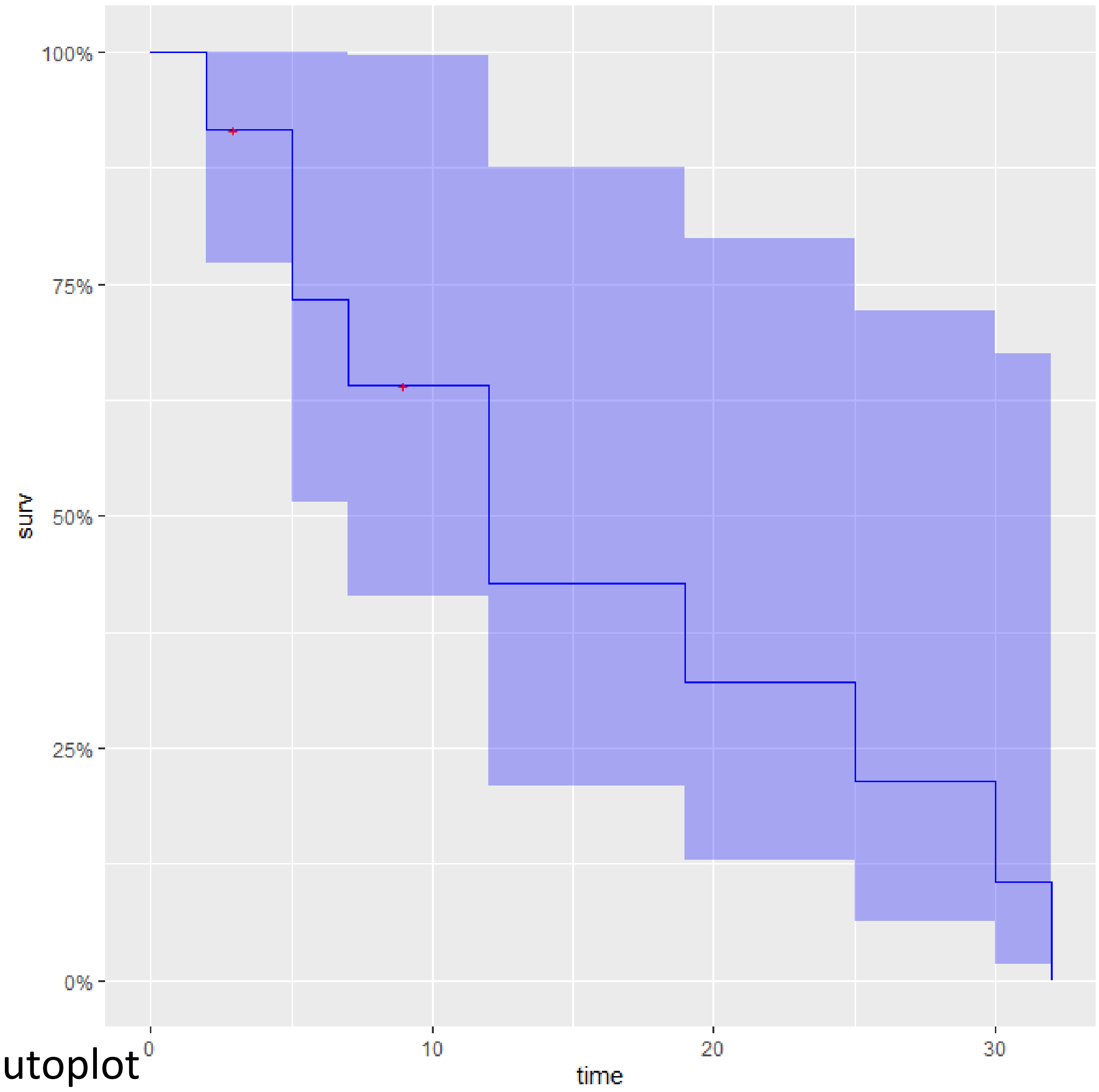
Kaplan-Meier Survival Analysis

time	status	*S*(*t*)
0		1
2	1	0.917
3	0	
5	1	
5	1	0.733
7	1	0.642
9	0	
12	1	
12	1	0.428
19	1	0.321
25	1	0.214
30	1	0.107
32	1	0

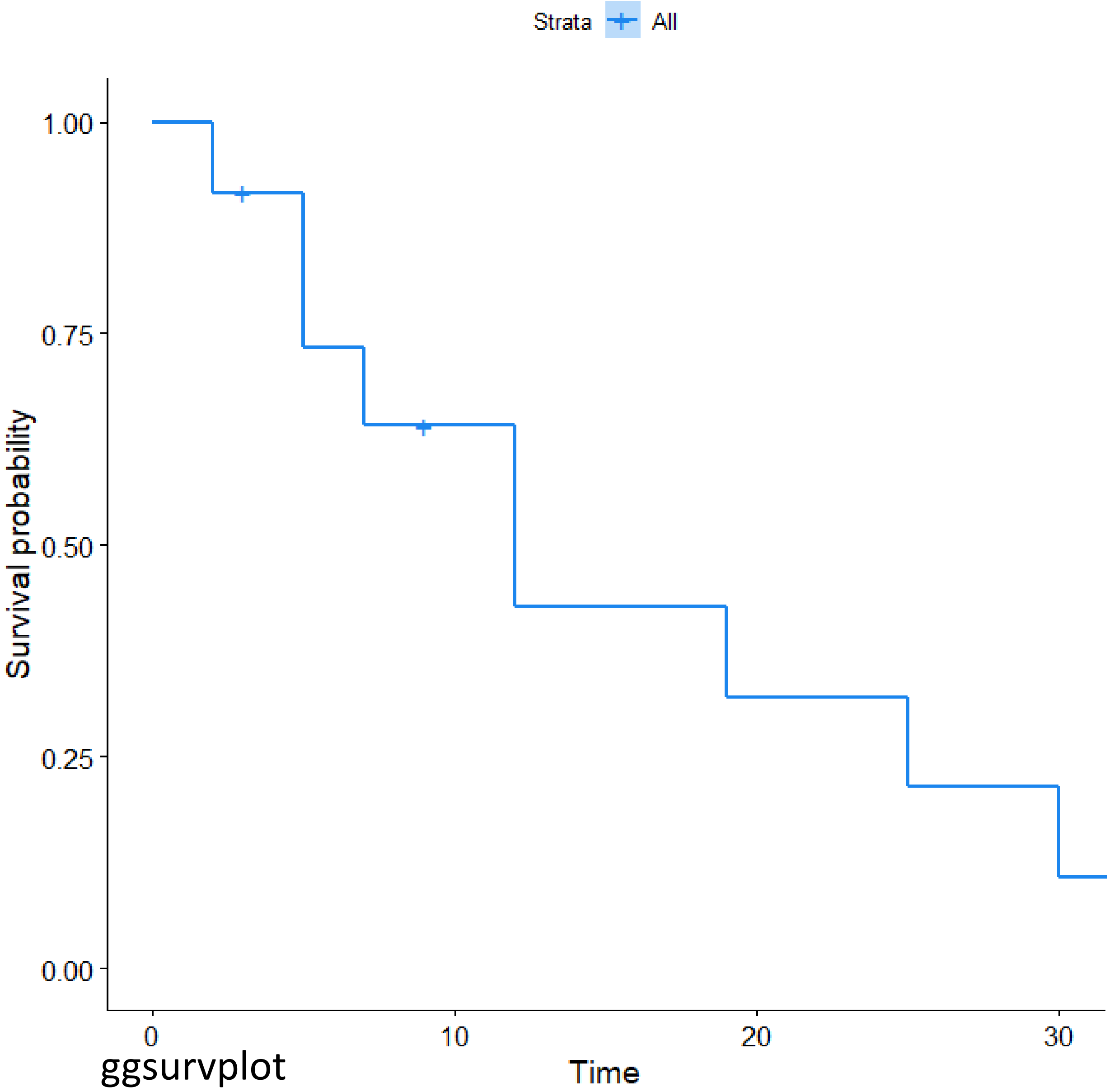
1 : dead
0 : censored



Kaplan-Meier Survival Analysis



autoplot

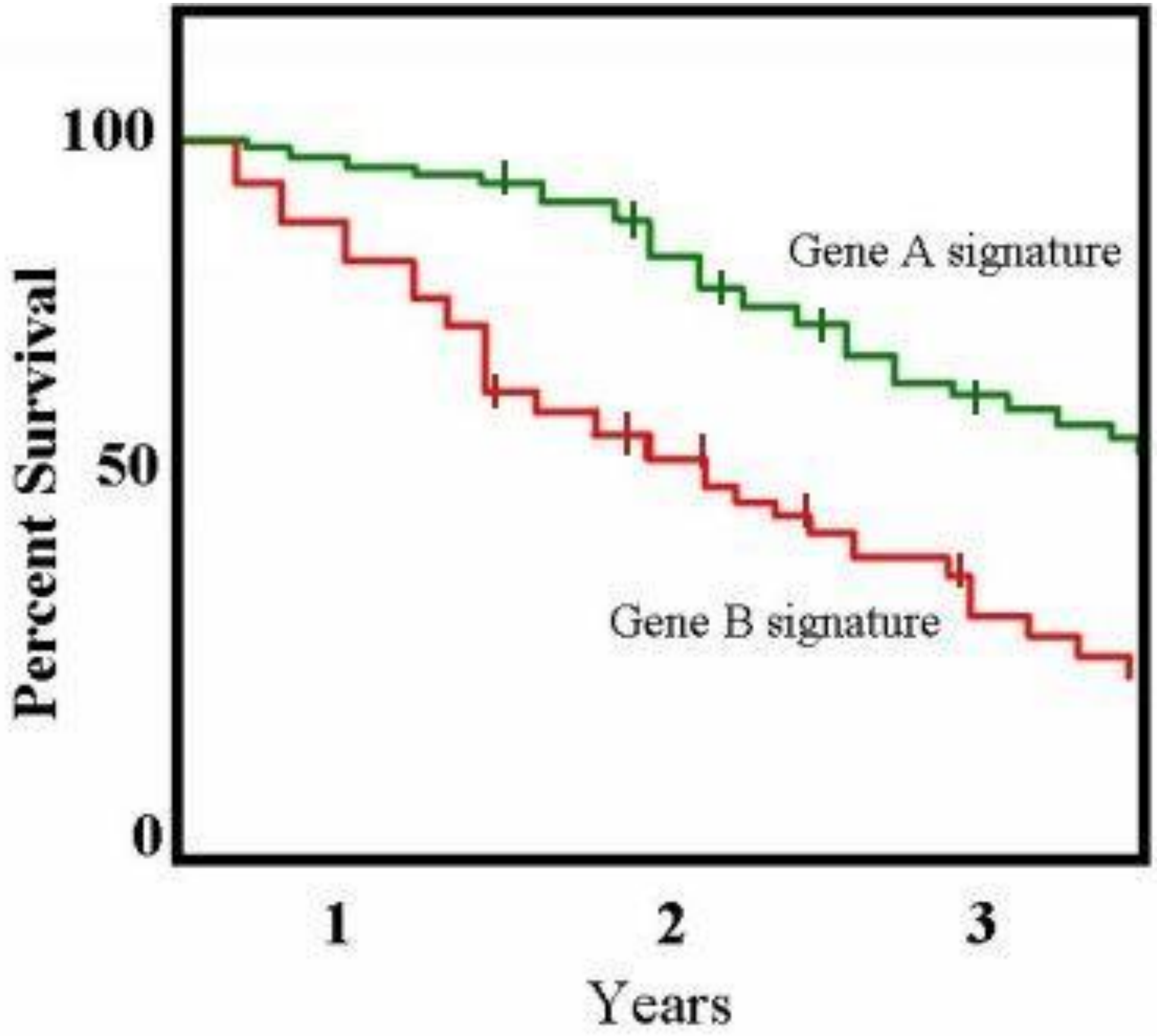


ggsurvplot

Kaplan-Meier Survival Analysis

```
#install.packages('ggfortify')
#install.packages('survival')
#install.packages('survminer')

library(ggfortify)
library(survival)
library(survminer)

data_KM<-read.table("C:/Users/seh00004.UCONN/Desktop/CSE5520/R/KM_data1.txt",sep="\t", header=TRUE)

fit <- survfit(Surv(time, status) ~ 1, data = data_KM)

autoplot(fit, surv.colour = 'blue', censor.colour = 'red', conf.int=FALSE)
dev.new()
autoplot(fit, surv.colour = 'blue', censor.colour = 'red', conf.colour = 'orange')

dev.new()
ggsurvplot(fit, palette=c("dodgerblue2"))

data_KM<-read.table("C:/Users/seh00004.UCONN/Desktop/CSE5520/R/KM_data1.txt",sep="\t", header=TRUE)
sfit <- survfit(Surv(time, status) ~ sex, data = data_KM)

dev.new()
ggsurvplot(sfit)

dev.new()
ggsurvplot(sfit, conf.int=TRUE, pval=TRUE, risk.table=TRUE,
        legend.labs=c("Male", "Female"), legend.title="Sex",
        palette=c("dodgerblue2", "orchid2"),
        title="Kaplan-Meier Curve",
        risk.table.height=.2)
survdiff(Surv(time, status)~sex, data=data_KM)
```

Kaplan-Meier Survival Analysis for 2 groups



Kaplan-Meier Survival Analysis for 2 groups

- The two survival curves can be compared statistically by testing the null hypothesis
- There is no difference regarding survival among two interventions.
- log-rank test :
  - E_1 and E_2 : the expected number of events in each group.
  - O_1 and O_2 : the total number of observed events in each group,
- The test statistic is

$$\frac{(0_1 - E_1)^2}{E_1} + \frac{(0_2 - E_2)^2}{E_2}$$

Kaplan-Meier Survival Analysis for 2 groups

- The total number of expected events in a group : the sum of expected number of events, at the time of each event in any of the group, taking both groups together.
  - At the time of event in any group the expected number of events is the product of risk of event at that time with the total number of subjects alive at the start of the time of event in that very group
  - e.g. at day 6, 46 patients were alive at the start of the day and one died, so the risk of event was 1/46 = 0.021739. As 23 patients were alive at the start of the day in group 2, the expected number of events at day 6 in group 2 was 23 × 0.021739 = 0.5.
  - The total number of expected events in group 2 is sum of the expected events calculated at different time.
- The test statistic and the significance can be drawn by comparing the calculated value with the critical value (using chi-square table) for degree of freedom equal to one.

Kaplan-Meier Survival Analysis for 2 groups

Table 3: Log-rank statistic for patients mentioned in examples 1 and 2

Time of event (t)	Total no. of patients died in both group (D)	No. of patients died in group 2 (O2)	Live at the start of the day (N)	Live at the start of the day in group 2 (n2)	Probability of death at the end of time (L)	Expected probability of death in group 2 (E2)	Expected probability of death in group 1 (E1)
6	1	0	46	23	0.021739	0.5	
9	1	1	45	23	0.022222	0.511111	
12	1	0	44	22	0.022727	0.5	
13	1	1	43	22	0.023256	0.511628	
21	1	0	42	21	0.02381	0.5	
27	2	1	40	21	0.05	1.05	
32	1	0	39	20	0.025641	0.512821	
38	1	1	38	20	0.026316	0.526316	
39	1	0	37	19	0.027027	0.513514	
43	2	0	36	19	0.055556	1.055556	
49	2	2	32	18	0.0625	1.125	
89	1	0	31	16	0.032258	0.516129	
93	1	1	29	15	0.034483	0.517241	
126	1	1	25	12	0.04	0.48	
218	1	1	19	9	0.052632	0.473684	
261	1	0	17	8	0.058824	0.470588	
263	1	0	15	7	0.066667	0.466667	
270	1	0	14	7	0.071429	0.5	
301	1	1	11	6	0.090909	0.545455	
311	1	0	10	5	0.1	0.5	
333	1	1	9	4	0.111111	0.444444	
	24	11				12.22015	11.77985

$$\text{Log-rank test statistic } = \frac{(0_1 - E_1)^2}{E_1} + \frac{(0_2 - E_2)^2}{E_2}$$

$$= \frac{(13 - 11.78)^2}{11.77} + \frac{(11 - 12.22)^2}{12.22}$$

$$= 0.1263 + 0.1218 = 0.2481$$

Manish Kumar Goel, Pardeep Khanna, Jugal Kishore, Understanding survival analysis: Kaplan-Meier estimate

Kaplan-Meier Survival Analysis for 2 groups

	P										
DF	0.995	0.975	0.2	0.1	0.05	0.025	0.02	0.01	0.005	0.002	0.001
1	.0004	.00016	1.642	2.706	3.841	5.024	5.412	6.635	7.879	9.55	10.828
2	0.01	0.0506	3.219	4.605	5.991	7.378	7.824	9.21	10.597	12.429	13.816
3	0.0717	0.216	4.642	6.251	7.815	9.348	9.837	11.345	12.838	14.796	16.266
4	0.207	0.484	5.989	7.779	9.488	11.143	11.668	13.277	14.86	16.924	18.467
5	0.412	0.831	7.289	9.236	11.07	12.833	13.388	15.086	16.75	18.907	20.515
6	0.676	1.237	8.558	10.645	12.592	14.449	15.033	16.812	18.548	20.791	22.458
7	0.989	1.69	9.803	12.017	14.067	16.013	16.622	18.475	20.278	22.601	24.322
8	1.344	2.18	11.03	13.362	15.507	17.535	18.168	20.09	21.955	24.352	26.124
9	1.735	2.7	12.242	14.684	16.919	19.023	19.679	21.666	23.589	26.056	27.877
10	2.156	3.247	13.442	15.987	18.307	20.483	21.161	23.209	25.188	27.722	29.588

$$\text{Log-rank test statistic } = \frac{(0_1 - E_1)^2}{E_1} + \frac{(0_2 - E_2)^2}{E_2}$$

$$= \frac{(13 - 11.78)^2}{11.77} + \frac{(11 - 12.22)^2}{12.22}$$

$$= 0.1263 + 0.1218 = 0.2481$$

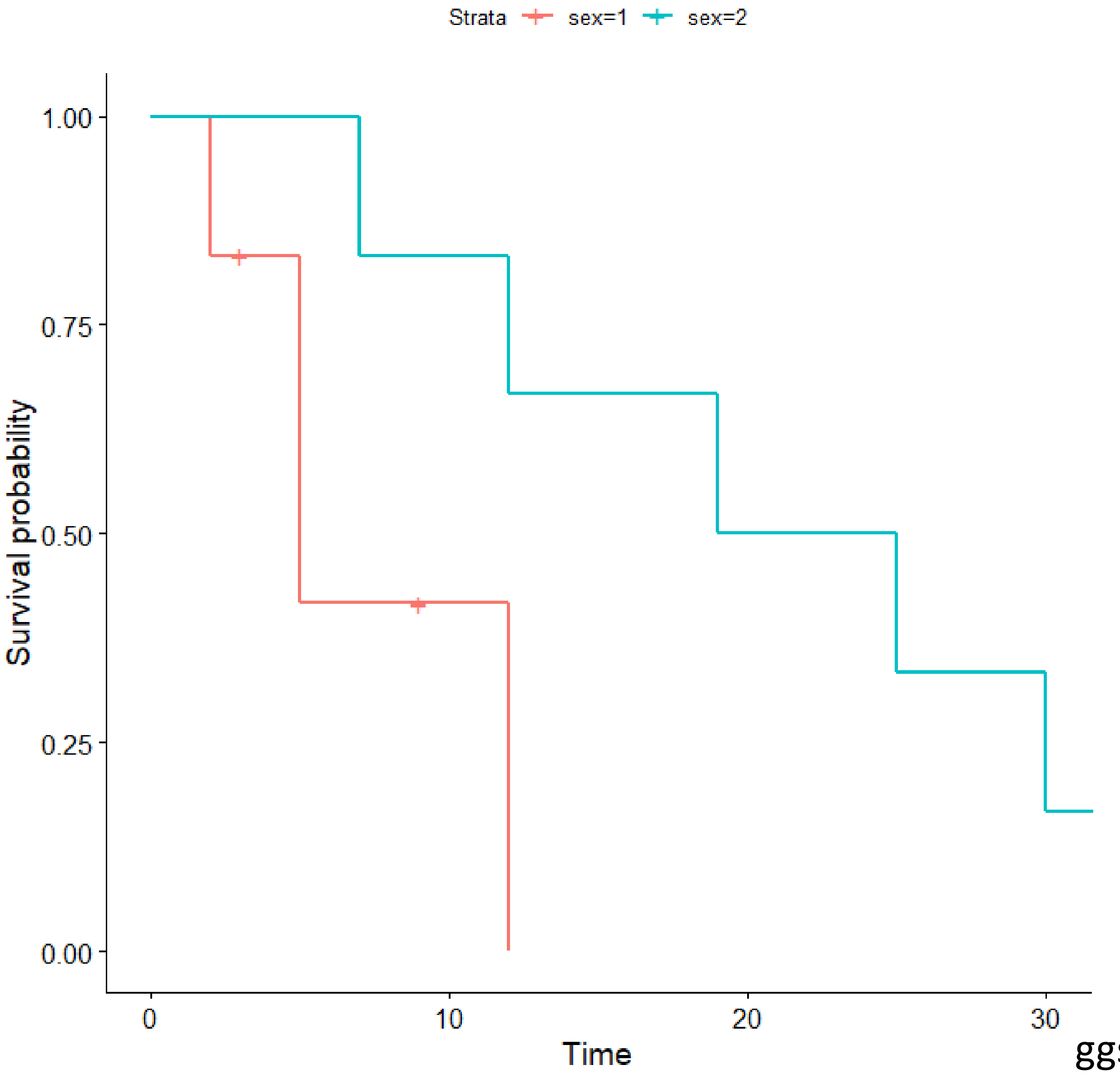
Kaplan-Meier Survival Analysis

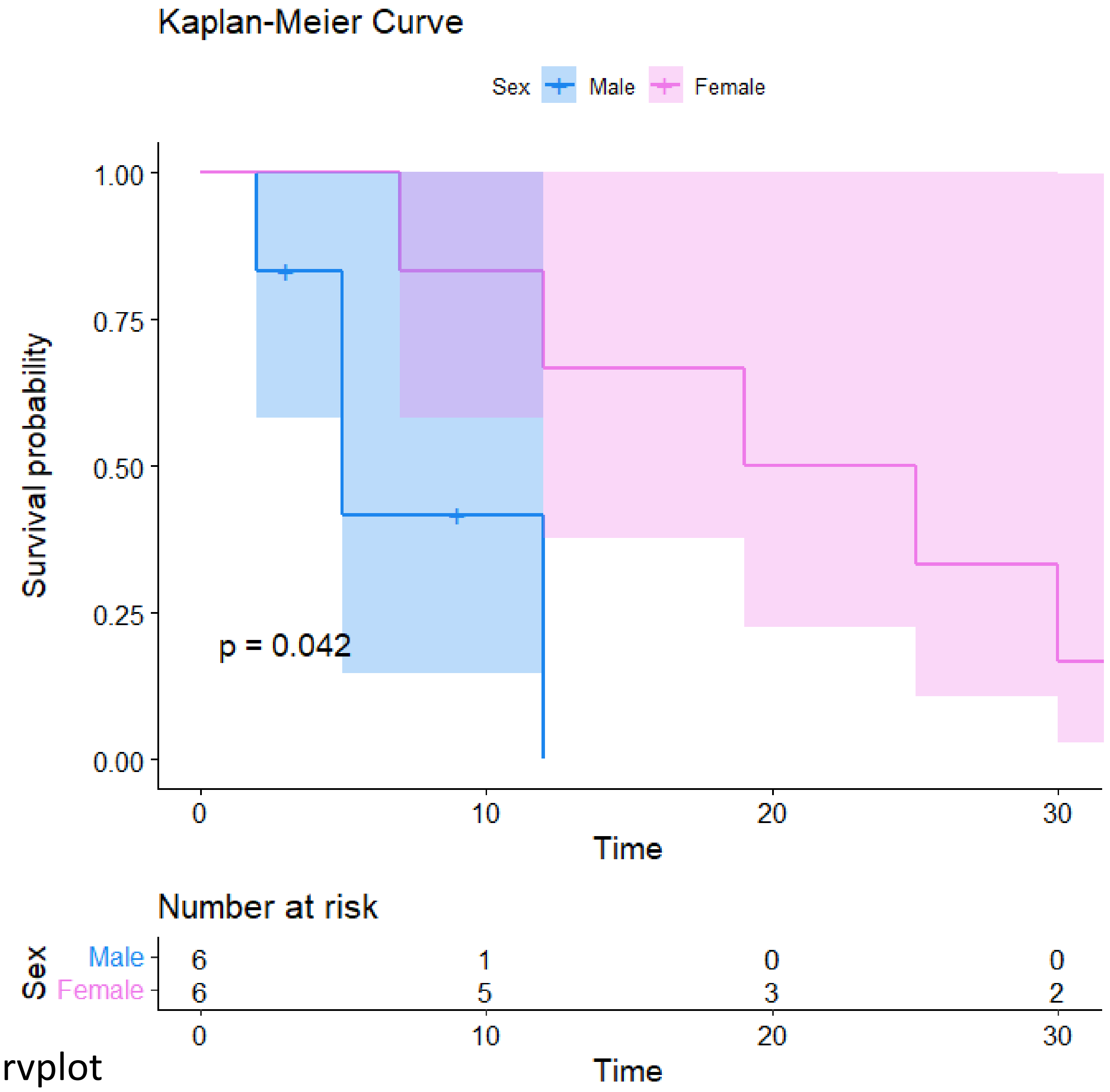
time	status	sex
2	1	1
3	0	1
5	1	1
5	1	1
7	1	2
9	0	1
12	1	2
12	1	1
19	1	2
25	1	2
30	1	2
32	1	2

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
sex=1	6	4	1.88	2.379	4.12
sex=2	6	6	8.12	0.552	4.12

Chisq= 4.1 on 1 degrees of freedom, p= 0.04

Kaplan-Meier Survival Analysis





ggsurvplot