



# **Statistics with R**

## *Survival Analysis*

**Deepayan Sarkar**

University of Wisconsin – Madison

*Summer Institute for Training in Biostatistics (2005)*

# Survival analysis in R

---

- Tools available in package `survival`
- This is a **recommended** package, which means it should already be installed
- It still has to be loaded using

```
> library(survival)
```
- Covered in Chapter 12 of the R text

# Functions of interest

---

- Create a survival object: `Surv`
- Kaplan-Meier estimates: `survfit`
- The log-rank test: `survdif`
- The Cox proportional hazards model: `coxph`  
(we won't be discussing this)

# Survival objects

---

- Created by the `Surv` function
- Needs two arguments:
  - `time`: follow-up time
  - `event`: status indicator
- `event=TRUE` means event occurred
- `event=FALSE` indicates censoring
- Other values possible (see `help(Surv)`)

# Example: melanom

---

We will use the example from the text:

```
> library(ISwR)
> data(melanom)
> str(melanom)
'data.frame':      205 obs. of  6 variables:
 $ no      : int  789 13 97 16 21 469 685 7 932 944 ...
 $ status: int  3 3 2 3 1 1 1 1 3 1 ...
 $ days   : int  10 30 35 99 185 204 210 232 232 279 ...
 $ ulc    : int  1 2 2 2 1 1 1 1 1 1 ...
 $ thick  : int  676 65 134 290 1208 484 516 1288 322 741 ...
 $ sex    : int  2 2 2 1 2 2 2 2 1 1 ...
```

We are interested in

- **days**: time on study after operation for malignant melanoma
- **status**: the patient's status at the end of study

# Censoring indicator

---

- The possible values of `status` are
  - 1: dead from malignant melanoma
  - 2: alive at end of study
  - 3: dead from other causes
- `Surv` needs a logical status indicator (`TRUE` if event occurred, `FALSE` is censored)
- Let's consider “dead from other causes” as censored
- Thus, status vector should be `status == 1`

# Creating the survival object

---

```
> msurv <- with(melanom, Surv(days, status == 1))
```

```
> msurv
```

```
 [1] 10+ 30+ 35+ 99+ 185 204 210 232 232+ 279 295 355+
[13] 386 426 469 493+ 529 621 629 659 667 718 752 779
[25] 793 817 826+ 833 858 869 872 967 977 982 1041 1055
[37] 1062 1075 1156 1228 1252 1271 1312 1427+ 1435 1499+ 1506 1508+
[49] 1510+ 1512+ 1516 1525+ 1542+ 1548 1557+ 1560 1563+ 1584 1605+ 1621
[61] 1627+ 1634+ 1641+ 1641+ 1648+ 1652+ 1654+ 1654+ 1667 1678+ 1685+ 1690
[73] 1710+ 1710+ 1726 1745+ 1762+ 1779+ 1787+ 1787+ 1793+ 1804+ 1812+ 1836+
[85] 1839+ 1839+ 1854+ 1856+ 1860+ 1864+ 1899+ 1914+ 1919+ 1920+ 1927+ 1933
[97] 1942+ 1955+ 1956+ 1958+ 1963+ 1970+ 2005+ 2007+ 2011+ 2024+ 2028+ 2038+
[109] 2056+ 2059+ 2061 2062 2075+ 2085+ 2102+ 2103 2104+ 2108 2112+ 2150+
[121] 2156+ 2165+ 2209+ 2227+ 2227+ 2256 2264+ 2339+ 2361+ 2387+ 2388 2403+
[133] 2426+ 2426+ 2431+ 2460+ 2467 2492+ 2493+ 2521+ 2542+ 2559+ 2565 2570+
[145] 2660+ 2666+ 2676+ 2738+ 2782 2787+ 2984+ 3032+ 3040+ 3042 3067+ 3079+
[157] 3101+ 3144+ 3152+ 3154+ 3180+ 3182+ 3185+ 3199+ 3228+ 3229+ 3278+ 3297+
[169] 3328+ 3330+ 3338 3383+ 3384+ 3385+ 3388+ 3402+ 3441+ 3458+ 3459+ 3459+
[181] 3476+ 3523+ 3667+ 3695+ 3695+ 3776+ 3776+ 3830+ 3856+ 3872+ 3909+ 3968+
[193] 4001+ 4103+ 4119+ 4124+ 4207+ 4310+ 4390+ 4479+ 4492+ 4668+ 4688+ 4926+
[205] 5565+
```

# Operations on the survival object

---

- Not very useful in isolation
- Typically used in other functions
- Fortunately, trying to naively compute the mean or median gives an error (although they are not very informative)

```
> mean(msurv)
```

```
Error in Summary.Surv(..., na.rm = na.rm) :  
  Invalid operation on a survival time
```

```
> median(msurv)
```

```
Error in "[.Surv"(sort(x, partial = c(half, half + 1)), c(half, half + 1)) :  
  subscript out of bounds
```

# The Kaplan-Meier estimator

---

- Computed by the function `survfit`
- Simplest case: just needs the survival object
- Note use of the `data` argument below

```
> mfit <- survfit(Surv(days, status == 1), data = melanom)
> mfit
Call: survfit(formula = Surv(days, status == 1), data = melanom)
```

```
      n  events  median 0.95LCL 0.95UCL
    205     57     Inf     Inf     Inf
```

```
> options(survfit.print.mean = TRUE)
```

```
> mfit
```

```
Call: survfit(formula = Surv(days, status == 1), data = melanom)
```

```
      n  events  rmean se(rmean)  median 0.95LCL 0.95UCL
    205     57   4125     161     Inf     Inf     Inf
```

# The Kaplan-Meier estimator (contd)

---

- The `print` method gives a very brief description
- The `summary` method actually produces the values of  $S$

```
> summary(mfit, times = seq(185, 3000, 400))
```

```
Call: survfit(formula = Surv(days, status == 1), data = melanom)
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
185	201	1	0.995	0.00496	0.985	1.000
585	188	9	0.950	0.01542	0.920	0.981
985	171	16	0.869	0.02397	0.823	0.917
1385	162	9	0.823	0.02713	0.772	0.878
1785	127	10	0.769	0.03033	0.712	0.831
2185	83	5	0.729	0.03358	0.666	0.798
2585	61	4	0.689	0.03729	0.620	0.766
2985	54	1	0.677	0.03854	0.605	0.757

- By default, values of  $S$  at all event times are listed
- Naturally enough, the `plot` method plots it

# The Kaplan-Meier estimator (contd)

---

```
> summary(mfit)
```

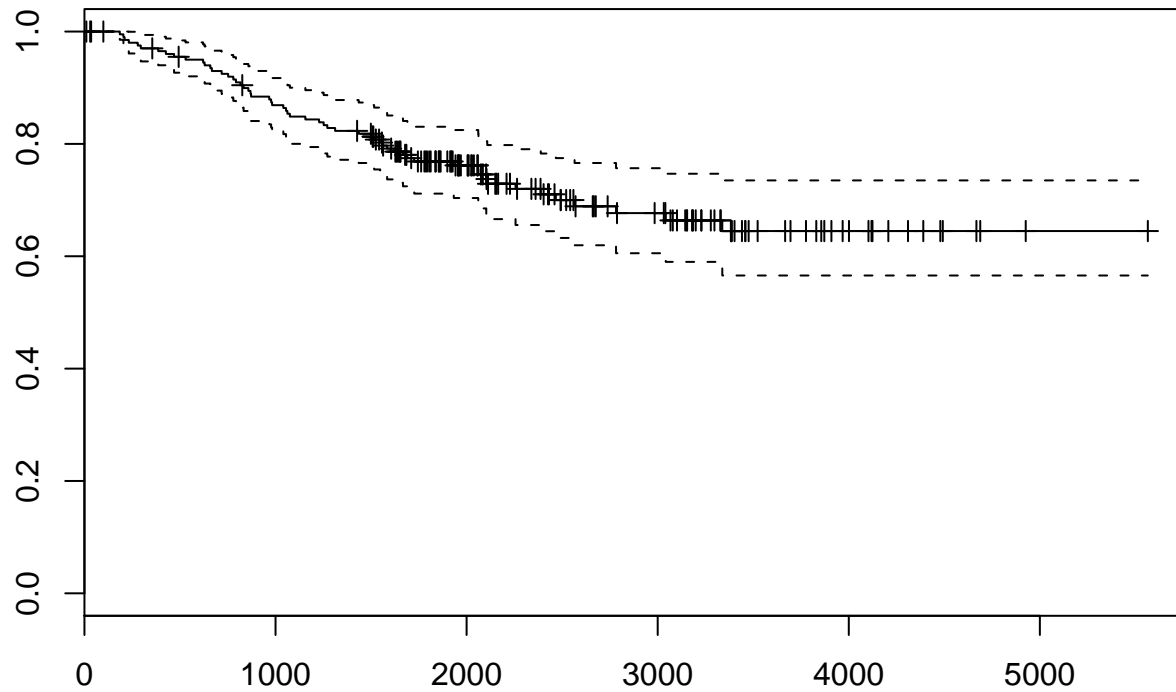
```
Call: survfit(formula = Surv(days, status == 1), data = melanom)
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
185	201	1	0.995	0.00496	0.985	1.000
204	200	1	0.990	0.00700	0.976	1.000
210	199	1	0.985	0.00855	0.968	1.000
232	198	1	0.980	0.00985	0.961	1.000
279	196	1	0.975	0.01100	0.954	0.997
295	195	1	0.970	0.01202	0.947	0.994
386	193	1	0.965	0.01297	0.940	0.991
426	192	1	0.960	0.01384	0.933	0.988
469	191	1	0.955	0.01465	0.927	0.984
529	189	1	0.950	0.01542	0.920	0.981
621	188	1	0.945	0.01615	0.914	0.977
629	187	1	0.940	0.01683	0.907	0.973
659	186	1	0.935	0.01748	0.901	0.970
667	185	1	0.930	0.01811	0.895	0.966
718	184	1	0.925	0.01870	0.889	0.962
752	183	1	0.920	0.01927	0.883	0.958
779	182	1	0.915	0.01981	0.877	0.954
793	181	1	0.910	0.02034	0.871	0.950
817	180	1	0.904	0.02084	0.865	0.946
833	178	1	0.899	0.02134	0.859	0.942
858	177	1	0.894	0.02181	0.853	0.938
869	176	1	0.889	0.02227	0.847	0.934
872	175	1	0.884	0.02272	0.841	0.930

# The Kaplan-Meier estimator (contd)

---

```
> plot(mfit)
```



# Groups

---

- Things get interesting when there are two or more groups to compare
- For example, does survival differ in men and women?

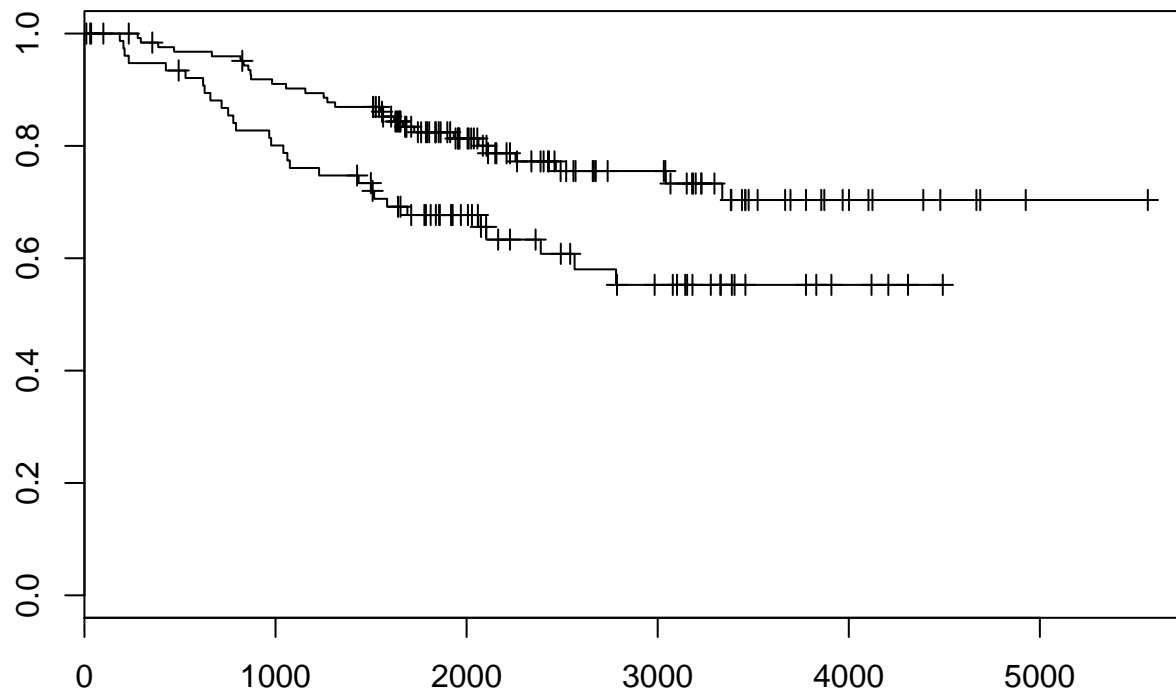
```
> mfit.bysex <- survfit(Surv(days, status == 1) ~ sex, data = melanom)
> mfit.bysex
Call: survfit(formula = Surv(days, status == 1) ~ sex, data = melanom)
```

	n	events	rmean	se(rmean)	median	0.95LCL	0.95UCL
sex=1	126	28	4416	191	Inf	Inf	Inf
sex=2	79	29	3065	207	Inf	2388	Inf

# Groups (contd)

---

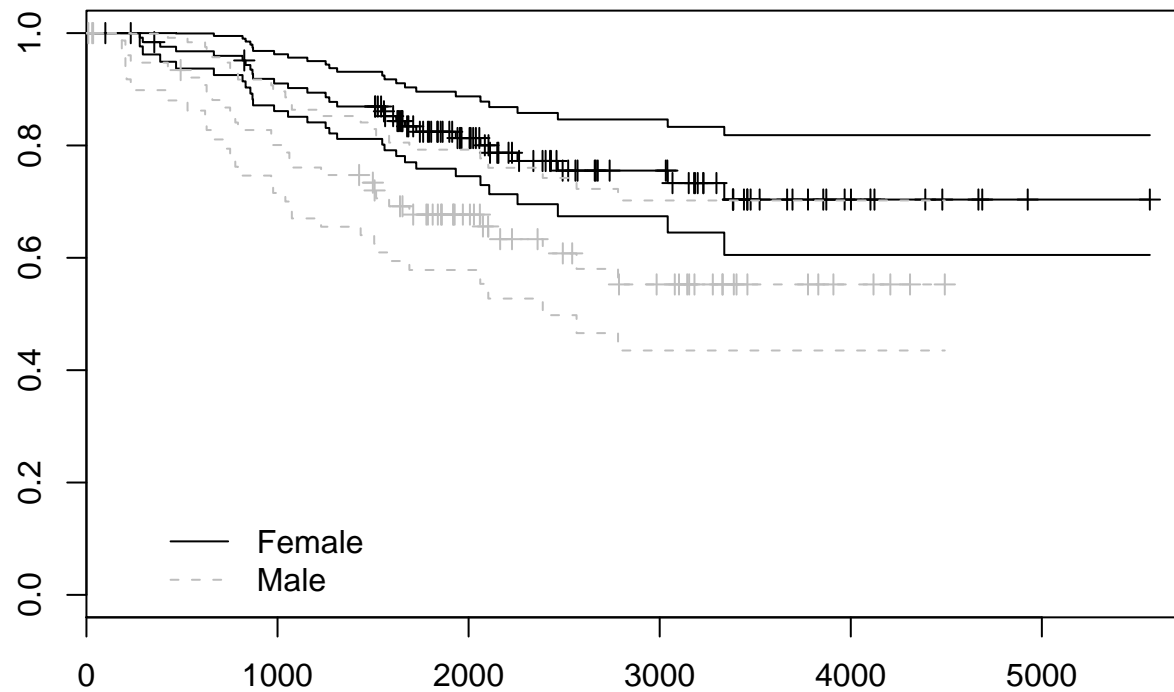
```
> plot(mfit.bysex)
```



# Groups (contd)

---

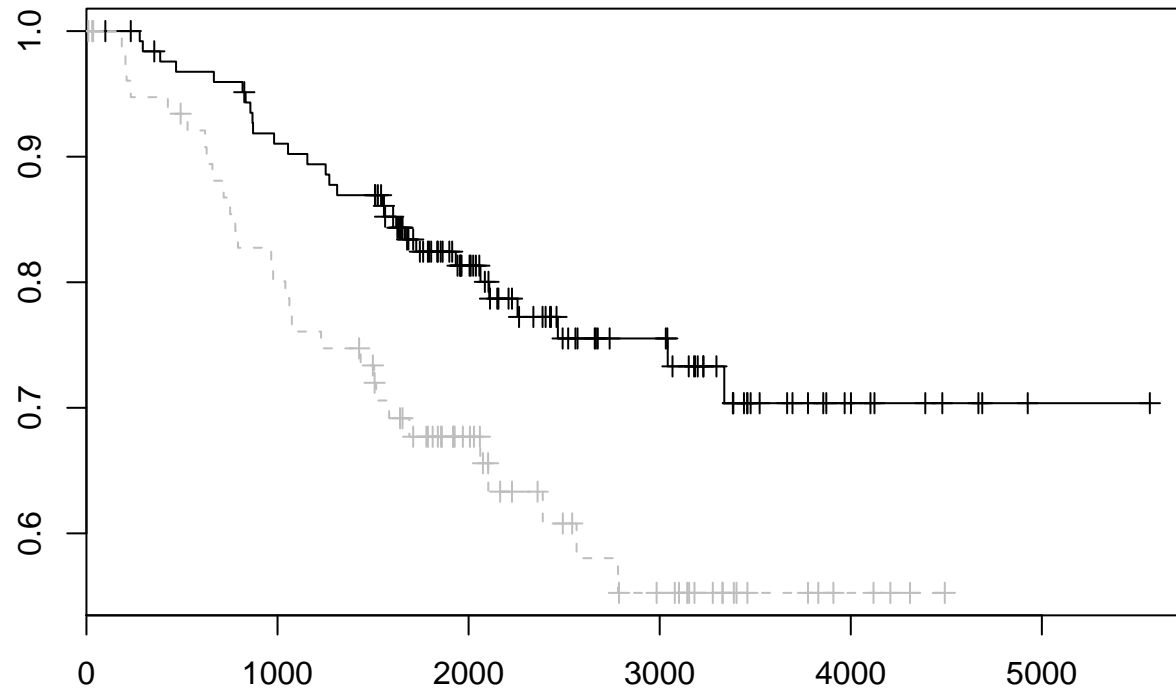
```
> plot(mfit.bysex, conf.int = TRUE, col = c("black",  
+     "grey"), lty = 1:2, legend.text = c("Female",  
+     "Male"))
```



# Transformations

---

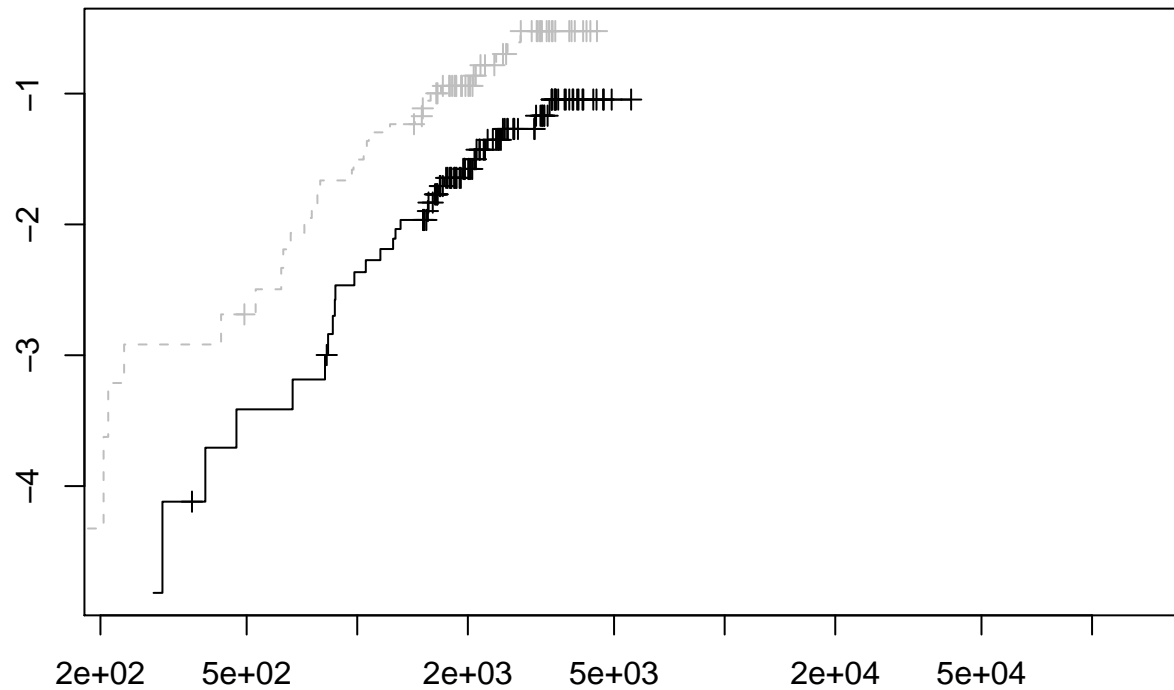
```
> plot(mfit.bysex, fun = "log", col = c("black",  
+   "grey"), lty = 1:2, legend.text = c("Female",  
+   "Male"))
```



# Transformations

---

```
> plot(mfit.bysex, fun = "cloglog", col = c("black",  
+     "grey"), lty = 1:2, legend.text = c("Female",  
+     "Male"))
```



# The log-rank test

---

Formally testing for differences between groups

```
> survdiff(Surv(days, status == 1) ~ sex, data = melanom)
```

```
Call:
```

```
survdiff(formula = Surv(days, status == 1) ~ sex, data = melanom)
```

	N	Observed	Expected	(O-E) <sup>2</sup> /E	(O-E) <sup>2</sup> /V
sex=1	126	28	37.1	2.25	6.47
sex=2	79	29	19.9	4.21	6.47

```
Chisq= 6.5 on 1 degrees of freedom, p= 0.011
```