```r
library(MASS)
library(tidyverse)
library(marginaleffects)
library(survival)
library(survminer)
```

Figure 1: Packages loaded

```r
GlowWorms %>% slice_sample(n = 10)
```

Lantern	Eggs
5.9	30
12.0	71
10.2	41
14.9	99
11.9	103
13.5	68
7.8	46
8.5	82
12.5	56
11.8	114

```r
GlowWorms %>% summarize(n = n(), mean_Lantern = mean(Lantern), mean_Eggs = mean(Eggs))
```

n	mean_Lantern	mean_Eggs
26	11.9	78.19231

Figure 2: Glowworms data (some randomly chosen rows), with summary statistics (for whole dataset)

```r
glow.1 <- lm(Eggs ~ Lantern, data = GlowWorms)
summary(glow.1)
```

```

Call:
lm(formula = Eggs ~ Lantern, data = GlowWorms)

Residuals:
   Min     1Q Median     3Q    Max 
-69.50 -23.59  -3.20  22.95  63.33 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -8.977     21.869  -0.410 0.685087    
Lantern        7.325      1.757   4.169 0.000343 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 32.71 on 24 degrees of freedom
Multiple R-squared:  0.4201,	Adjusted R-squared:  0.3959 
F-statistic: 17.38 on 1 and 24 DF,  p-value: 0.0003431
```

Figure 3: Glowworms regression

```r
new <- datagrid(model = glow.1, Lantern = c(5, 12, 19))
cbind(predictions(glow.1, newdata = new)) %>%
  select(estimate, conf.low, conf.high, Lantern)
```

estimate	conf.low	conf.high	Lantern
27.64885	0.768477	54.52923	5
78.92482	66.348540	91.50110	12
130.20079	102.709804	157.69178	19

Figure 4: Glowworms predictions 1

```r
p <- predict(glow.1, new, interval = "p")
cbind(new, p)
```

Eggs	Lantern	fit	lwr	upr
78	5	27.64885	-45.54768	100.8454
78	12	78.92482	10.13605	147.7136
78	19	130.20079	56.75322	203.6484

Figure 5: Glowworms predictions 2, using same `new` as in previous Figure

```
heart
```

mck	ha	nha
20	2	88
60	13	26
100	30	8
140	30	5
180	21	0
220	19	1
260	18	1
300	13	1
340	19	0
380	15	0
420	7	0
460	8	0
500	35	0

Figure 6: Creatinine kinase data

```
heart %>% select(ends_with("ha")) %>%
  as.matrix() -> response
response
```

```
      ha nha
 [1,]  2  88
 [2,] 13  26
 [3,] 30   8
 [4,] 30   5
 [5,] 21   0
 [6,] 19   1
 [7,] 18   1
 [8,] 13   1
 [9,] 19   0
[10,] 15   0
[11,]  7   0
[12,]  8   0
[13,] 35   0
```

Figure 7: Creatinine kinase data: some code and its output

```
heart.1 <- glm(response ~ mck, data = heart, family = "binomial")
summary(heart.1)
```

```
Call:
glm(formula = response ~ mck, family = "binomial", data = heart)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.028360   0.366977  -8.252   <2e-16 ***
mck          0.035104   0.004081   8.602   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 311.29  on 12  degrees of freedom
Residual deviance:  28.14  on 11  degrees of freedom
AIC: 51.596

Number of Fisher Scoring iterations: 6
```

Figure 8: Creatinine kinase logistic regression

```
new <- tibble(mck = seq(0, 300, 20))
cbind(predictions(heart.1, newdata = new)) %>%
  select(estimate, conf.low, conf.high, mck) %>%
  ggplot(aes(x = mck, y = estimate, ymin = conf.low, ymax = conf.high)) +
  geom_line()  + geom_ribbon(alpha = 0.2)
```

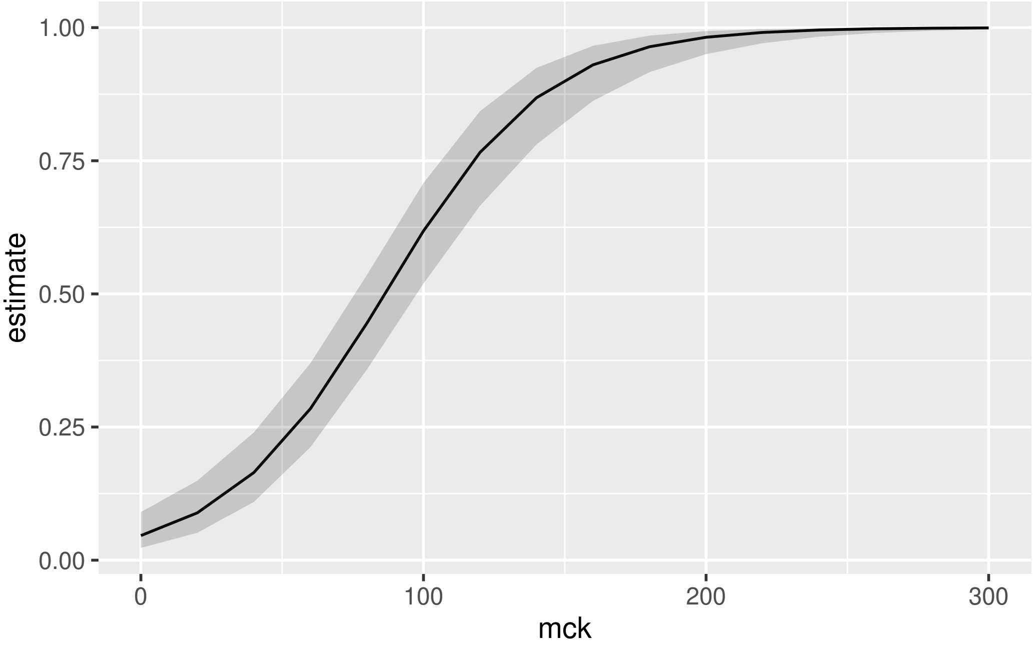

Figure 9: Creatinine kinase model predictions

```
new2 <- tibble(mck = c(20, 40, 60, 80))
cbind(predictions(heart.1, newdata = new2, type = "link")) %>%
  select(estimate, mck)
```

estimate	mck
-2.3262717	20
-1.6241839	40
-0.9220960	60
-0.2200082	80

Figure 10: Creatinine kinase data: more predictions

```
Faces %>% slice_sample(n = 20)
```

MaxGripStrength	SHR	Attractive
42.0	1.249	A
49.5	1.284	A
58.5	1.428	D
44.0	1.402	C
39.0	1.269	A
49.5	1.643	D
57.0	1.431	A
46.5	1.363	D
43.5	1.249	A
49.0	1.136	A
44.0	1.352	B
34.0	1.307	B
66.0	1.545	D
37.0	1.377	C
45.5	1.436	C
48.0	1.405	B
37.0	1.368	C
41.0	1.322	D
44.5	1.333	B
50.5	1.218	A

Figure 11: Attractiveness data (random sample of rows)

```
Faces.1 <- polr(Attractive ~ MaxGripStrength + SHR + I(SHR^2), data = Faces)
```

Figure 12: Model for attractiveness data

```
drop1(Faces.1, test = "Chisq")
```

	Df	AIC	LRT	Pr(>Chi)
	NA	102.8388	NA	NA
MaxGripStrength	1	100.9271	0.088272	0.7663855
SHR	1	102.4722	1.633377	0.2012366
I(SHR^2)	1	102.9592	2.120328	0.1453554

Figure 13: `drop1` output for attractiveness data model

```
new <- datagrid(model = Faces.1, SHR = c(1, 1.2, 1.4, 1.6), MaxGripStrength = c(42, 54))
cbind(predictions(Faces.1, newdata = new)) %>%
  select(group, estimate, SHR, MaxGripStrength) %>%
  pivot_wider(names_from = group, values_from = estimate)
```

```
Re-fitting to get Hessian
```

SHR	MaxGripStrength	A	B	C	D
1.0	42	0.2037797	0.3097473	0.2532528	0.2332202
1.0	54	0.1952400	0.3049208	0.2569195	0.2429197
1.2	42	0.5097292	0.3011724	0.1194417	0.0696567
1.2	54	0.4963609	0.3062037	0.1242323	0.0732031
1.4	42	0.1388698	0.2605816	0.2749903	0.3255583
1.4	54	0.1325973	0.2540969	0.2758976	0.3374082
1.6	42	0.0009542	0.0029696	0.0081966	0.9878797
1.6	54	0.0009045	0.0028157	0.0077763	0.9885035

Figure 14: Predictions of attractiveness for various values of shoulder-hip ratio and grip strength

```r
larynx %>% slice(1:20)
```

stage	time	age	diagyr	delta
stage1	0.6	77	76	dead
stage1	1.3	53	71	dead
stage1	2.4	45	71	dead
stage1	2.5	57	78	alive
stage1	3.2	58	74	dead
stage1	3.2	51	77	alive
stage1	3.3	76	74	dead
stage1	3.3	63	77	alive
stage1	3.5	43	71	dead
stage1	3.5	60	73	dead
stage1	4.0	52	71	dead
stage1	4.0	63	76	dead
stage1	4.3	86	74	dead
stage1	4.5	48	76	alive
stage1	4.5	68	76	alive
stage1	5.3	81	72	dead
stage1	5.5	70	75	alive
stage1	5.9	58	75	alive
stage1	5.9	47	75	alive
stage1	6.0	75	73	dead

Figure 15: Larynx cancer data (some)

```
 [1] 0.6  1.3  2.4  2.5+ 3.2  3.2+ 3.3  3.3+ 3.5  3.5  4.0  4.0  4.3  4.5+ 4.5+
[16] 5.3  5.5+ 5.9+ 5.9+ 6.0
```

Figure 16: Larynx cancer: some values of `y`

```
larynx.1 <- coxph(y ~ stage + age + diagyr, data = larynx)
summary(larynx.1)
```

```
Call:
coxph(formula = y ~ stage + age + diagyr, data = larynx)

  n= 90, number of events= 50 

                coef exp(coef) se(coef)      z Pr(>|z|)    
stagestage2  0.15164   1.16375  0.46481  0.326   0.7442    
stagestage3  0.64473   1.90546  0.35619  1.810   0.0703 .  
stagestage4  1.73211   5.65255  0.43596  3.973 7.09e-05 ***
age          0.01869   1.01887  0.01433  1.304   0.1922    
diagyr      -0.01819   0.98198  0.07646 -0.238   0.8120    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
stagestage2     1.164     0.8593    0.4680     2.894
stagestage3     1.905     0.5248    0.9480     3.830
stagestage4     5.653     0.1769    2.4052    13.284
age             1.019     0.9815    0.9906     1.048
diagyr          0.982     1.0184    0.8453     1.141

Concordance= 0.674  (se = 0.039 )
Likelihood ratio test= 18.37  on 5 df,   p=0.003
Wald test            = 21.2  on 5 df,   p=7e-04
Score (logrank) test = 24.84  on 5 df,   p=1e-04
```

```
drop1(larynx.1, test = "Chisq")
```

	Df	AIC	LRT	Pr(>Chi)
	NA	385.3583	NA	NA
stage	3	394.8735	15.515268	0.0014253
age	1	385.0995	1.741195	0.1869875
diagyr	1	383.4147	0.056456	0.8121876

Figure 17: Larynx cancer: Cox model 1

```r
larynx.2 <- coxph(y ~ stage, data = larynx)
summary(larynx.2)
```

```
Call:
coxph(formula = y ~ stage, data = larynx)

  n= 90, number of events= 50 

               coef exp(coef) se(coef)     z Pr(>|z|)    
stagestage2 0.06481   1.06696  0.45843 0.141   0.8876    
stagestage3 0.61481   1.84930  0.35519 1.731   0.0835 .  
stagestage4 1.73490   5.66838  0.41939 4.137 3.52e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
stagestage2     1.067     0.9372    0.4344      2.62
stagestage3     1.849     0.5407    0.9219      3.71
stagestage4     5.668     0.1764    2.4916     12.90

Concordance= 0.668  (se = 0.037 )
Likelihood ratio test= 16.49  on 3 df,   p=9e-04
Wald test            = 19.24  on 3 df,   p=2e-04
Score (logrank) test = 22.88  on 3 df,   p=4e-05
```

```r
drop1(larynx.2, test = "Chisq")
```

	Df	AIC	LRT	Pr(>Chi)
	NA	383.2416	NA	NA
stage	3	393.7270	16.48538	0.0009016

Figure 18: Larynx cancer: Cox model 2

```r
anova(larynx.2, larynx.1)
```

loglik	Chisq	Df	Pr(>|Chi|)
-188.6208	NA	NA	NA
-187.6791	1.883308	2	0.3899823

Figure 19: Larynx cancer: further analysis

```
larynx %>% count(stage) -> new
new
```

stage	n
stage1	33
stage2	17
stage3	27
stage4	13

```
s <- survfit(larynx.2, new, data = larynx)
ggsurvplot(s, conf.int = FALSE)
```

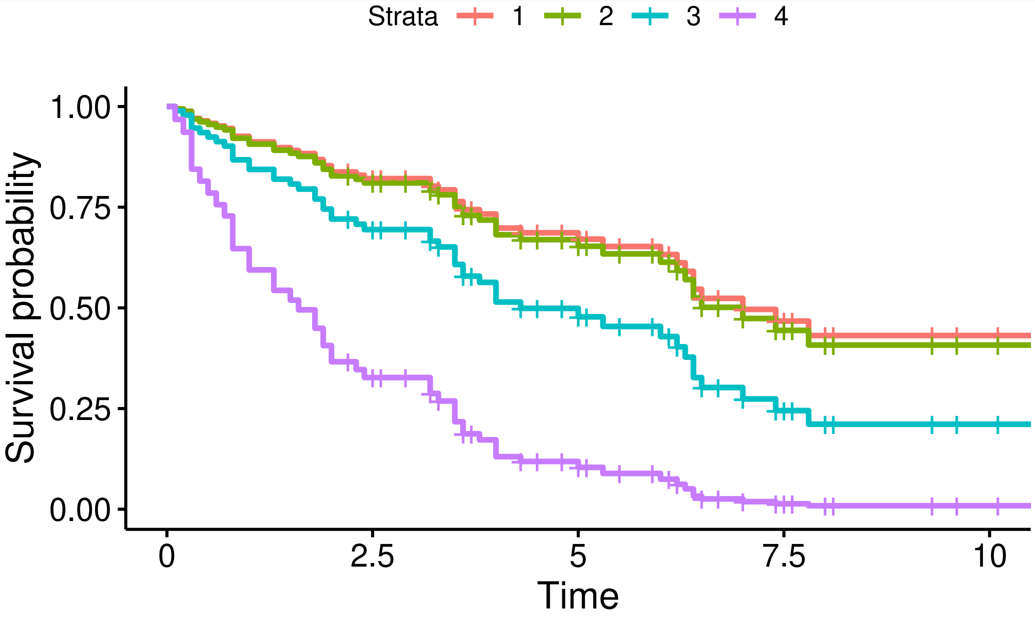

Figure 20: Larynx cancer: predicted survival curves

```
Wrinkle
```

Time	Fingers	Objects
73	non	dry
106	wrinkled	dry
85	wrinkled	wet
107	wrinkled	dry
94	non	dry
136	non	wet
108	wrinkled	wet
83	wrinkled	wet
96	non	dry
96	non	wet
95	wrinkled	wet
108	non	wet
97	non	dry
113	non	dry
109	wrinkled	wet
93	non	wet
143	non	wet
88	non	wet
135	wrinkled	wet
123	non	wet

Figure 21: Wrinkled fingers data (some randomly chosen rows)

```
Wrinkle %>%
  group_by(Fingers, Objects) %>%
  summarize(mean_time = mean(Time)) -> Wrinkle_means
```

```
`summarise()` has grouped output by 'Fingers'. You can override using the
`.groups` argument.
```

```
Wrinkle_means
```

Fingers	Objects	mean_time
non	dry	93.30
non	wet	117.90
wrinkled	dry	94.15
wrinkled	wet	102.85

```
ggplot(Wrinkle_means, aes(x = Objects, y = mean_time, colour = Fingers, group = Fingers)) +
  geom_point() + geom_line()
```

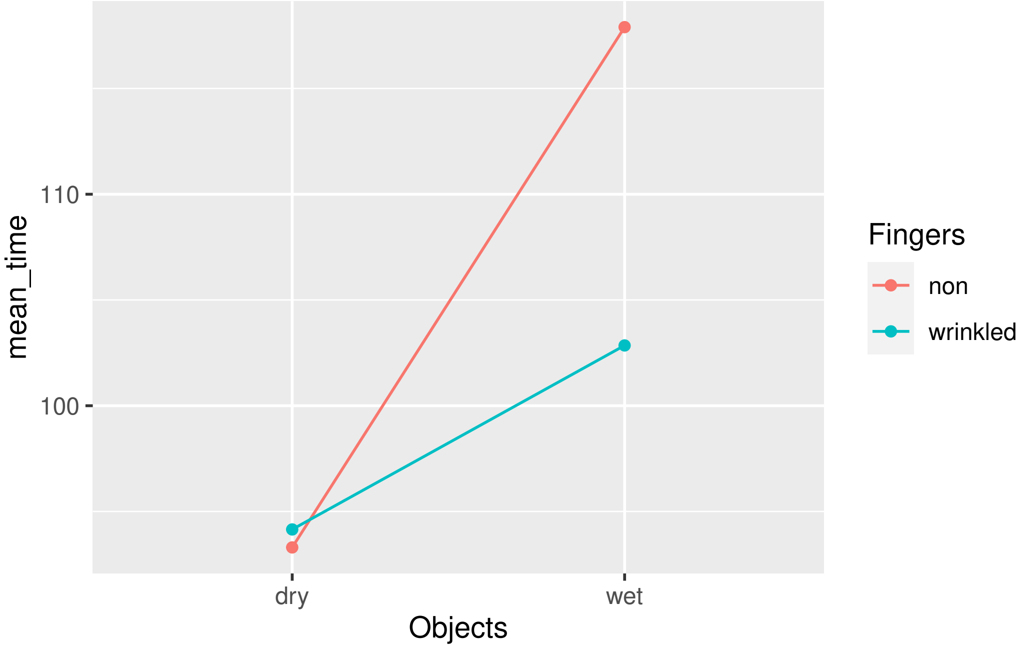

Figure 22: Wrinkled fingers plot

```
Wrinkle.1 <- aov(Time ~ Fingers * Objects, data = Wrinkle)
summary(Wrinkle.1)
```

```
                Df Sum Sq Mean Sq F value   Pr(>F)    
Fingers          1   1008    1008   2.447 0.121939    
Objects          1   5544    5544  13.454 0.000451 ***
Fingers:Objects  1   1264    1264   3.067 0.083912 .  
Residuals       76  31319     412                     
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 23: Wrinkled fingers analysis

```
Wrinkle %>%
  filter(Objects == "dry") -> drys
drys.1 <- aov(Time ~ Fingers, data = drys)
summary(drys.1)
```

```
            Df Sum Sq Mean Sq F value Pr(>F)
Fingers      1      7    7.23    0.03  0.863
Residuals   38   9035  237.76               
```

```
Wrinkle %>%
  filter(Objects == "wet") -> wets
wets.1 <- aov(Time ~ Fingers, data = wets)
summary(wets.1)
```

```
            Df Sum Sq Mean Sq F value Pr(>F)  
Fingers      1   2265  2265.0   3.862 0.0567 .
Residuals   38  22284   586.4                 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 24: Wrinkled fingers analysis continued