

Logistic regression

Predict binary outcomes (success/failure) from numerical or categorical predictors.

Linear vs. logistic regression

Linear regression:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

Linear vs. logistic regression

Linear regression:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

Logistic regression:

$$\Pr(\textit{success}) = \frac{e^t}{1 + e^t}$$

$$t = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

Linear vs. logistic regression

Linear regression:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

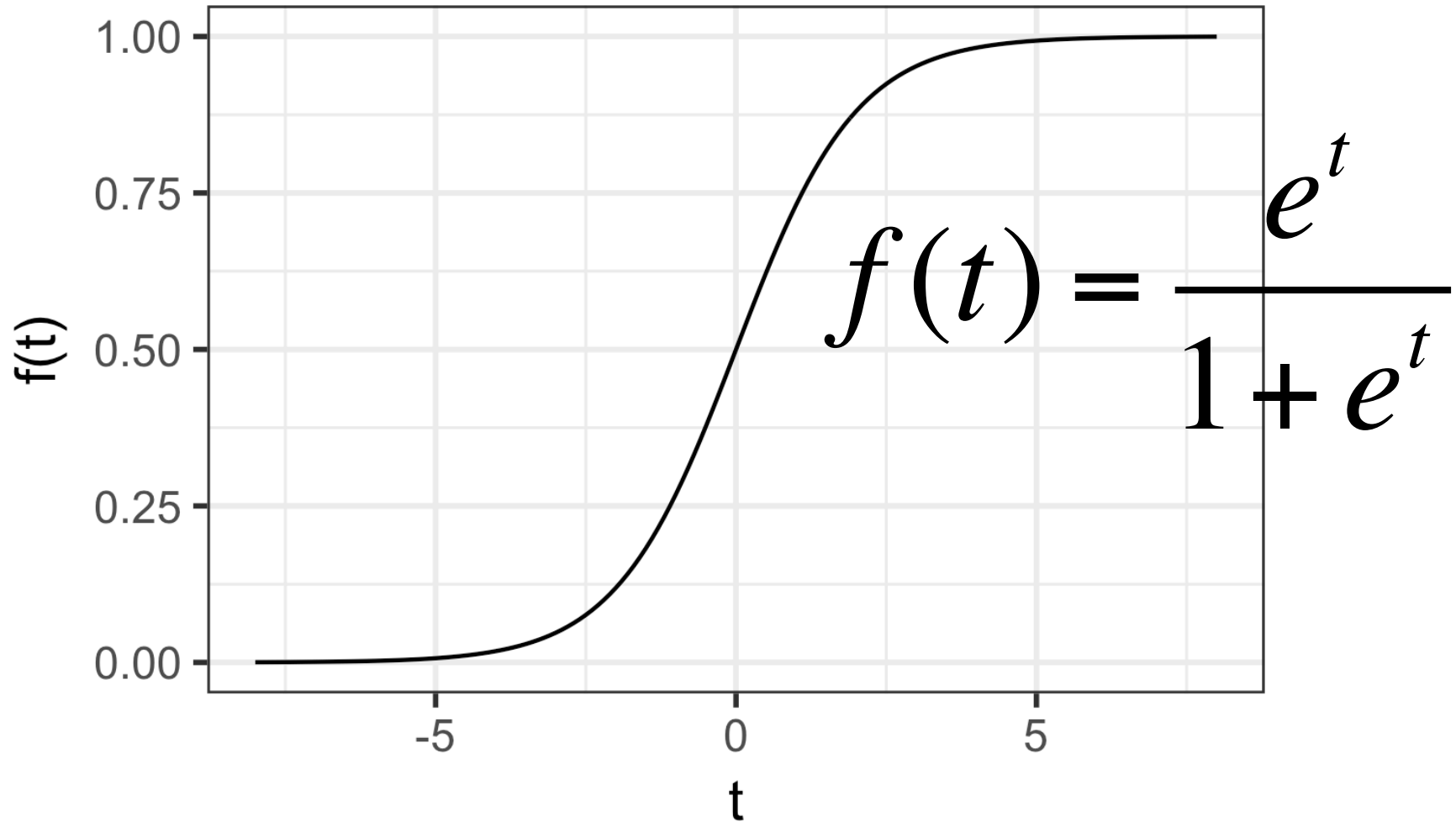
Logistic regression:

$$\Pr(\textit{success}) = \frac{e^t}{1 + e^t}$$

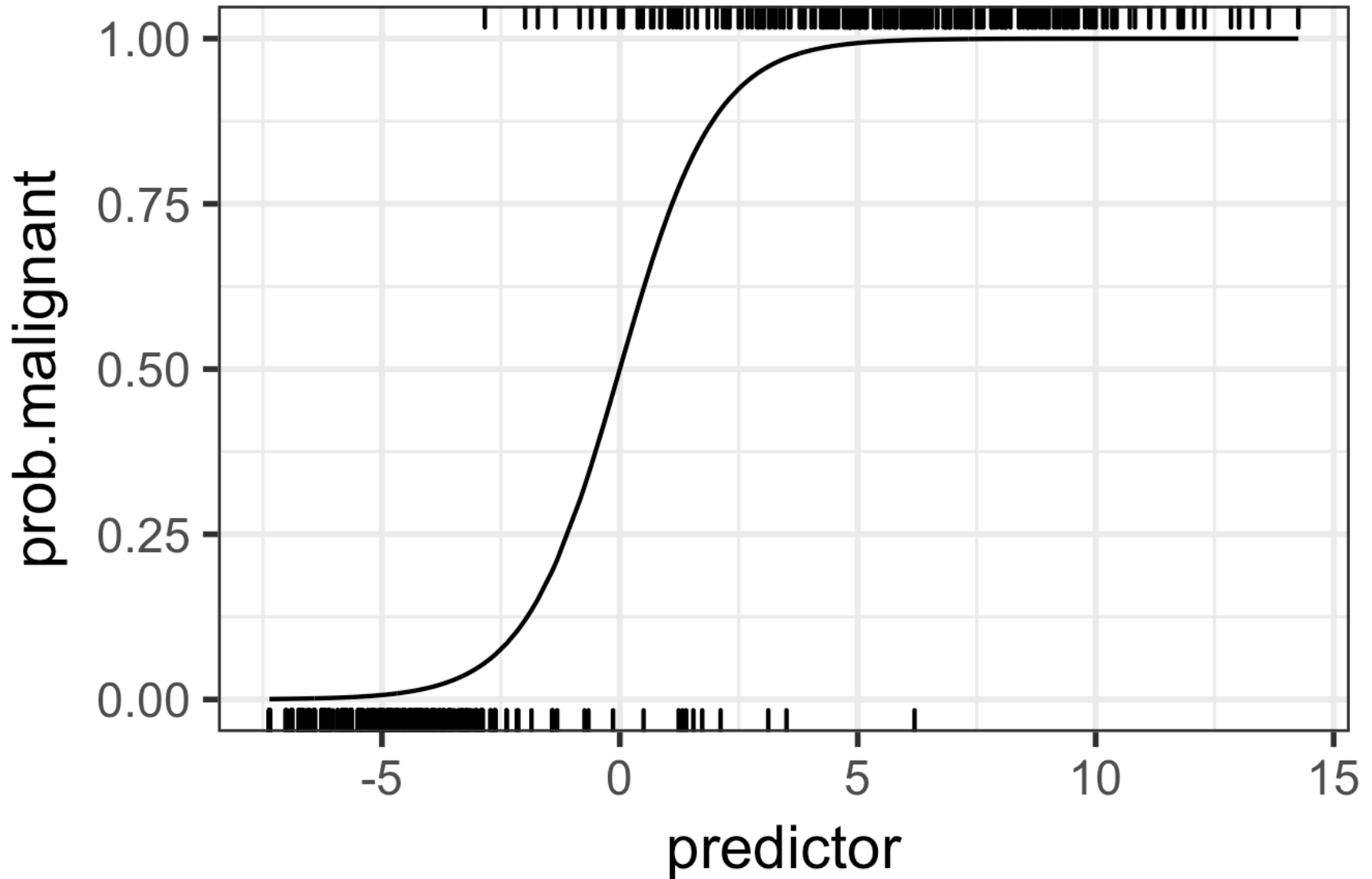
$$t = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon$$

(generalized linear model, GLM)

The logistic equation



Example: Pr(malignant) in biopsy data set



Let's do this step by step...

Recall the biopsy data set

```
> biopsy <- read.csv("http://wilkelab.org/classes/SDS348/data_sets/biopsy.csv")
> head(biopsy)
  clump_thickness uniform_cell_size uniform_cell_shape marg_adhesion
1                5                 1                 1                1
2                5                 4                 4                5
3                3                 1                 1                1
4                6                 8                 8                1
5                4                 1                 1                3
6                8                10                10                8
  epithelial_cell_size bare_nuclei bland_chromatin normal_nucleoli mitoses
1                    2             1              3                1         1
2                    7            10             3                2         1
3                    2             2              3                1         1
4                    3             4              3                7         1
5                    2             1              3                1         1
6                    7            10             9                7         1
  outcome
1   benign
2   benign
3   benign
4   benign
5   benign
6 malignant
```


We do logistic regression with the `glm()` function

```
> glm.out <- glm(outcome ~ clump_thickness +  
                  uniform_cell_size +  
                  uniform_cell_shape +  
                  marg_adhesion +  
                  epithelial_cell_size +  
                  bare_nuclei +  
                  bland_chromatin +  
                  normal_nucleoli +  
                  mitoses,  
                  data=biopsy,  
                  family=binomial)
```

```
> summary(glm.out)
```

```
Call:
```

```
glm(formula = outcome ~ clump_thickness + uniform_cell_size +  
    uniform_cell_shape + marg_adhesion + epithelial_cell_size +  
    bare_nuclei + bland_chromatin + normal_nucleoli + mitoses,  
    family = binomial, data = biopsy)
```

```
Deviance Residuals:
```

Min	1Q	Median	3Q	Max
-3.4841	-0.1153	-0.0619	0.0222	2.4698

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-10.10394	1.17488	-8.600	< 2e-16	***
clump_thickness	0.53501	0.14202	3.767	0.000165	***
uniform_cell_size	-0.00628	0.20908	-0.030	0.976039	
uniform_cell_shape	0.32271	0.23060	1.399	0.161688	
marg_adhesion	0.33064	0.12345	2.678	0.007400	**
epithelial_cell_size	0.09663	0.15659	0.617	0.537159	
bare_nuclei	0.38303	0.09384	4.082	4.47e-05	***
bland_chromatin	0.44719	0.17138	2.609	0.009073	**
normal_nucleoli	0.21303	0.11287	1.887	0.059115	.
mitoses	0.53484	0.32877	1.627	0.103788	

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> summary(glm.out)
```

Call:

```
glm(formula = outcome ~ clump_thickness + uniform_cell_size +  
    uniform_cell_shape + marg_adhesion + epithelial_cell_size +  
    bare_nuclei + bland_chromatin + normal_nucleoli + mitoses,  
    family = binomial, data = biopsy)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.4841	-0.1153	-0.0619	0.0222	2.4698

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-10.10394	1.17488	-8.600	< 2e-16	***
clump_thickness	0.53501	0.14202	3.767	0.000165	***
uniform_cell_size	0.00628	0.20908	0.030	0.976039	
uniform_cell_shape	0.32271	0.23060	1.399	0.161688	
marg_adhesion	0.33064	0.12345	2.678	0.007400	**
epithelial_cell_size	0.09663	0.15659	0.617	0.537159	
bare_nuclei	0.38303	0.09384	4.082	4.47e-05	***
bland_chromatin	0.44719	0.17138	2.609	0.009073	**
normal_nucleoli	0.21303	0.11287	1.887	0.059115	.
mitoses	0.53484	0.32877	1.627	0.103788	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> glm.out <- glm(outcome ~ clump_thickness +  
                  uniform_cell_shape +  
                  marg_adhesion +  
                  epithelial_cell_size +  
                  bare_nuclei +  
                  bland_chromatin +  
                  normal_nucleoli +  
                  mitoses,  
                  data=biopsy,  
                  family=binomial)
```

```
> summary(glm.out)
```

Call:

```
glm(formula = outcome ~ clump_thickness + uniform_cell_shape +  
    marg_adhesion + epithelial_cell_size + bare_nuclei +  
bland_chromatin +  
    normal_nucleoli + mitoses, family = binomial, data = biopsy)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.4823	-0.1154	-0.0620	0.0222	2.4694

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-10.09765	1.15546	-8.739	< 2e-16	***
clump_thickness	0.53456	0.14125	3.784	0.000154	***
uniform_cell_shape	0.31816	0.17424	1.826	0.067847	.
marg_adhesion	0.32993	0.12115	2.723	0.006465	**
epithelial_cell_size	0.09612	0.15564	0.618	0.536876	
bare_nuclei	0.38308	0.09384	4.082	4.46e-05	***
bland_chromatin	0.44648	0.16986	2.628	0.008578	**
normal_nucleoli	0.21255	0.11174	1.902	0.057149	.
mitoses	0.53406	0.32761	1.630	0.103064	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> summary(glm.out)
```

```
Call:
```

```
glm(formula = outcome ~ clump_thickness + uniform_cell_shape +  
    marg_adhesion + epithelial_cell_size + bare_nuclei +  
bland_chromatin +  
    normal_nucleoli + mitoses, family = binomial, data = biopsy)
```

```
Deviance Residuals:
```

Min	1Q	Median	3Q	Max
-3.4823	-0.1154	-0.0620	0.0222	2.4694

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-10.09765	1.15546	-8.739	< 2e-16	***
clump_thickness	0.53456	0.14125	3.784	0.000154	***
uniform_cell_shape	0.31816	0.17424	1.826	0.067847	.
marg_adhesion	0.32993	0.12115	2.723	0.006465	**
epithelial_cell_size	0.09612	0.15564	0.618	0.536876	
bare_nuclei	0.38308	0.09384	4.082	4.46e-05	***
bland_chromatin	0.44648	0.16986	2.628	0.008578	**
normal_nucleoli	0.21255	0.11174	1.902	0.057149	.
mitoses	0.53406	0.32761	1.630	0.103064	

```
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> glm.out <- glm(outcome ~ clump_thickness +  
                  uniform_cell_shape +  
                  marg_adhesion +  
                  bare_nuclei +  
                  bland_chromatin +  
                  normal_nucleoli +  
                  mitoses,  
                  data=biopsy,  
                  family=binomial)
```

```
> summary(glm.out)
```

Call:

```
glm(formula = outcome ~ clump_thickness + uniform_cell_shape +  
    marg_adhesion + bare_nuclei + bland_chromatin +  
normal_nucleoli +  
    mitoses, family = binomial, data = biopsy)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.5235	-0.1149	-0.0627	0.0219	2.4115

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-9.98278	1.12610	-8.865	< 2e-16	***
clump_thickness	0.53400	0.14079	3.793	0.000149	***
uniform_cell_shape	0.34529	0.17164	2.012	0.044255	*
marg_adhesion	0.34249	0.11922	2.873	0.004068	**
bare_nuclei	0.38830	0.09356	4.150	3.32e-05	***
bland_chromatin	0.46194	0.16820	2.746	0.006025	**
normal_nucleoli	0.22606	0.11097	2.037	0.041644	*
mitoses	0.53119	0.32446	1.637	0.101598	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


```
> summary(glm.out)
```

```
Call:
```

```
glm(formula = outcome ~ clump_thickness + uniform_cell_shape +  
    marg_adhesion + bare_nuclei + bland_chromatin +  
normal_nucleoli +  
    mitoses, family = binomial, data = biopsy)
```

```
Deviance Residuals:
```

Min	1Q	Median	3Q	Max
-3.5235	-0.1149	-0.0627	0.0219	2.4115

```
Coefficients:
```

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-9.98278	1.12610	-8.865	< 2e-16	***
clump_thickness	0.53400	0.14079	3.793	0.000149	***
uniform_cell_shape	0.34529	0.17164	2.012	0.044255	*
marg_adhesion	0.34249	0.11922	2.873	0.004068	**
bare_nuclei	0.38830	0.09356	4.150	3.32e-05	***
bland_chromatin	0.46194	0.16820	2.746	0.006025	**
normal_nucleoli	0.22606	0.11097	2.037	0.041644	*
mitoses	0.53119	0.32446	1.637	0.101598	

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> glm.out <- glm(outcome ~ clump_thickness +  
                  uniform_cell_shape +  
                  marg_adhesion +  
                  bare_nuclei +  
                  bland_chromatin +  
                  normal_nucleoli,  
                  data=biopsy,  
                  family=binomial)
```

```
> summary(glm.out)
```

Call:

```
glm(formula = outcome ~ clump_thickness + uniform_cell_shape +  
    marg_adhesion + bare_nuclei + bland_chromatin +  
normal_nucleoli,  
    family = binomial, data = biopsy)
```

Deviance Residuals:

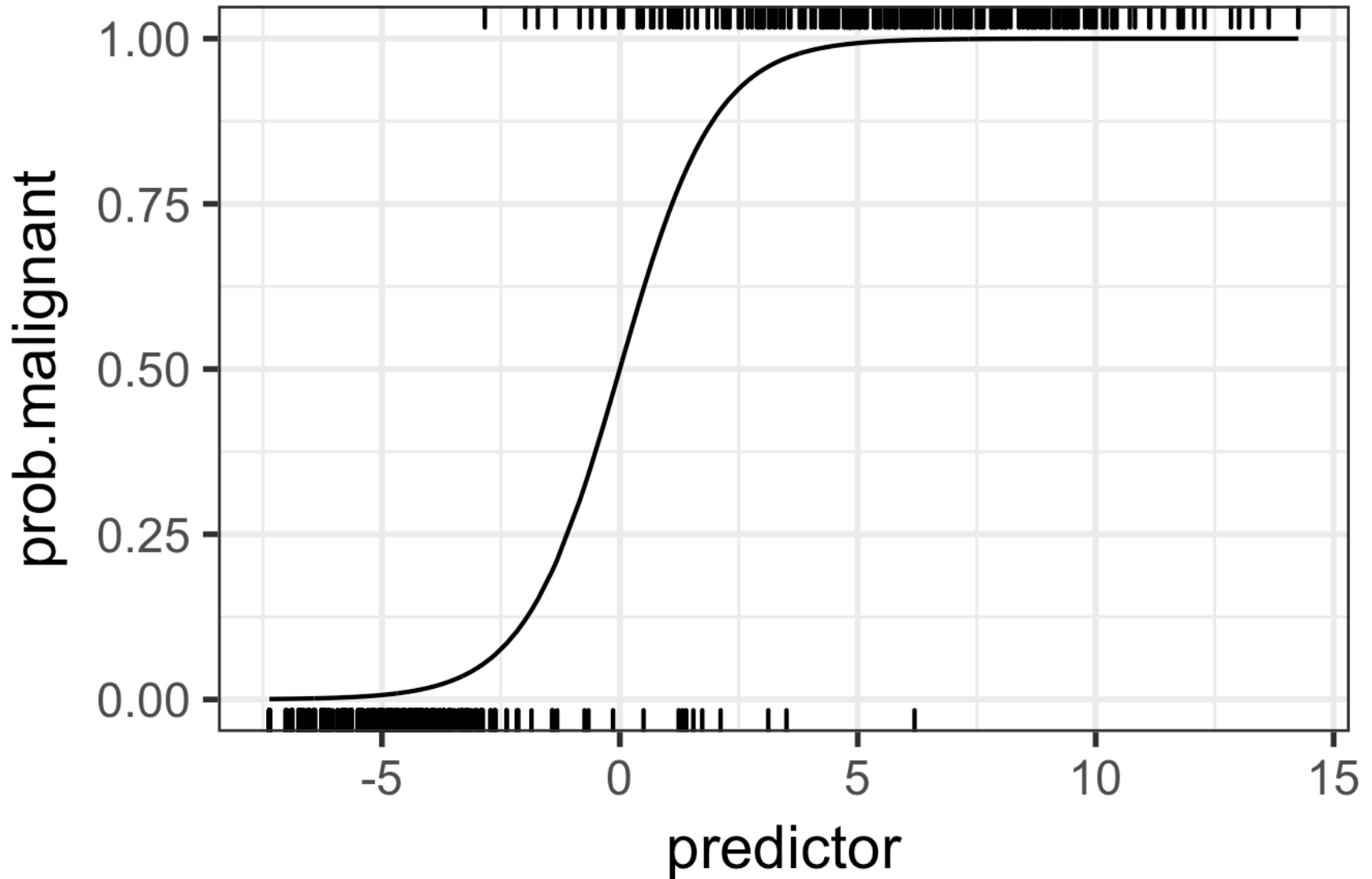
Min	1Q	Median	3Q	Max
-3.5201	-0.1186	-0.0570	0.0250	2.4055

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-9.76708	1.08506	-9.001	< 2e-16	***
clump_thickness	0.62253	0.13712	4.540	5.62e-06	***
uniform_cell_shape	0.34951	0.16503	2.118	0.03419	*
marg_adhesion	0.33753	0.11561	2.920	0.00350	**
bare_nuclei	0.37855	0.09381	4.035	5.45e-05	***
bland_chromatin	0.47134	0.16612	2.837	0.00455	**
normal_nucleoli	0.24317	0.10855	2.240	0.02509	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

The fitted logistic model



We can extract fitted probabilities from `glm.out$fitted.values`

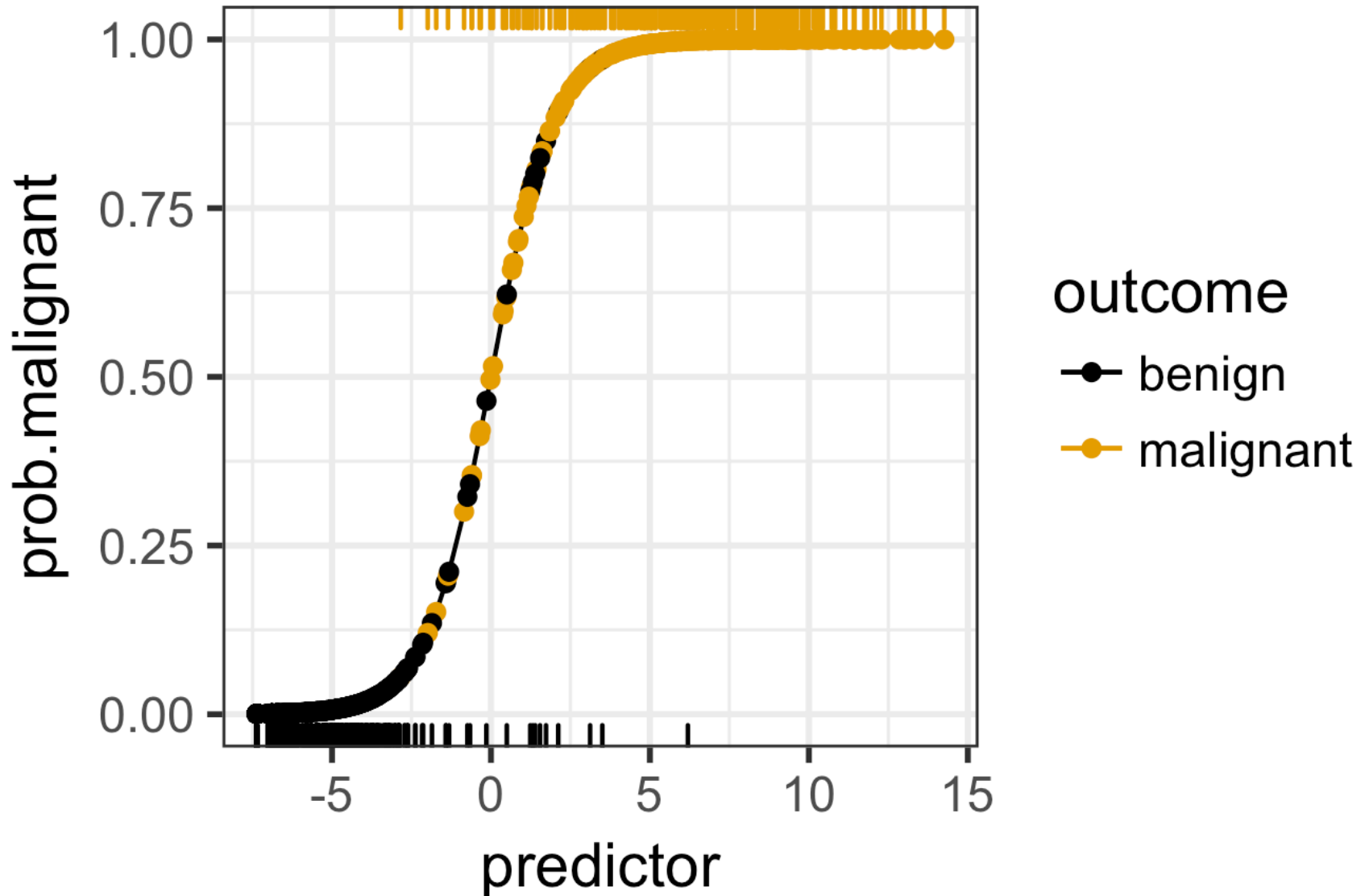
```
> glm.out$fitted.values
      1          2          3          4          5          6
0.0192341317 0.8925583864 0.0081774737 0.8496854505 0.0202506282 0.9999854554
      7          8          9         10         11         12
0.0467606911 0.0042790664 0.0011789931 0.0065253423 0.0016231293 0.0018875638
     13         14         15         16         17         18
0.3544332567 0.0034543023 0.9993353305 0.7371582761 0.0065253423 0.0104135504
     19         20         21         22         23         24
0.9989353409 0.0352597948 0.9969203982 0.9994994519 0.0035120154 0.0016231293
     25         26         27         28         29         30
0.7802514369 0.0035120154 0.0120927435 0.0018875638 0.0012725934 0.0035120154
     31         32         33         34         35         36
0.0030206952 0.9977220579 0.0042283384 0.0049740412 0.0018875638 0.9998755391
     37         38         39         40         41         42
0.1940709471 0.9954253327 0.6691128086 0.9536389392 0.9974078013 0.3002866244
     43         44         45         46         47         48
0.9996235802 0.0010137236 0.9583091930 0.0010137236 0.0202506282 0.9836985106
     49         50         51         52         53         54
0.7842860362 0.4122043566 0.9956800184 0.9922376046 0.9988895968 0.9870508267
     55         56         57         58         59         60
0.9927513406 0.6585108620 0.7534314353 0.8341431018 0.9032183182 0.0014795146
     61         62         63         64         65         66
0.9921570845 0.5158282353 0.0010137236 0.7040691331 0.0104135504 0.9498144607
```

We can extract linear predictors from `glm.out$linear.predictors`

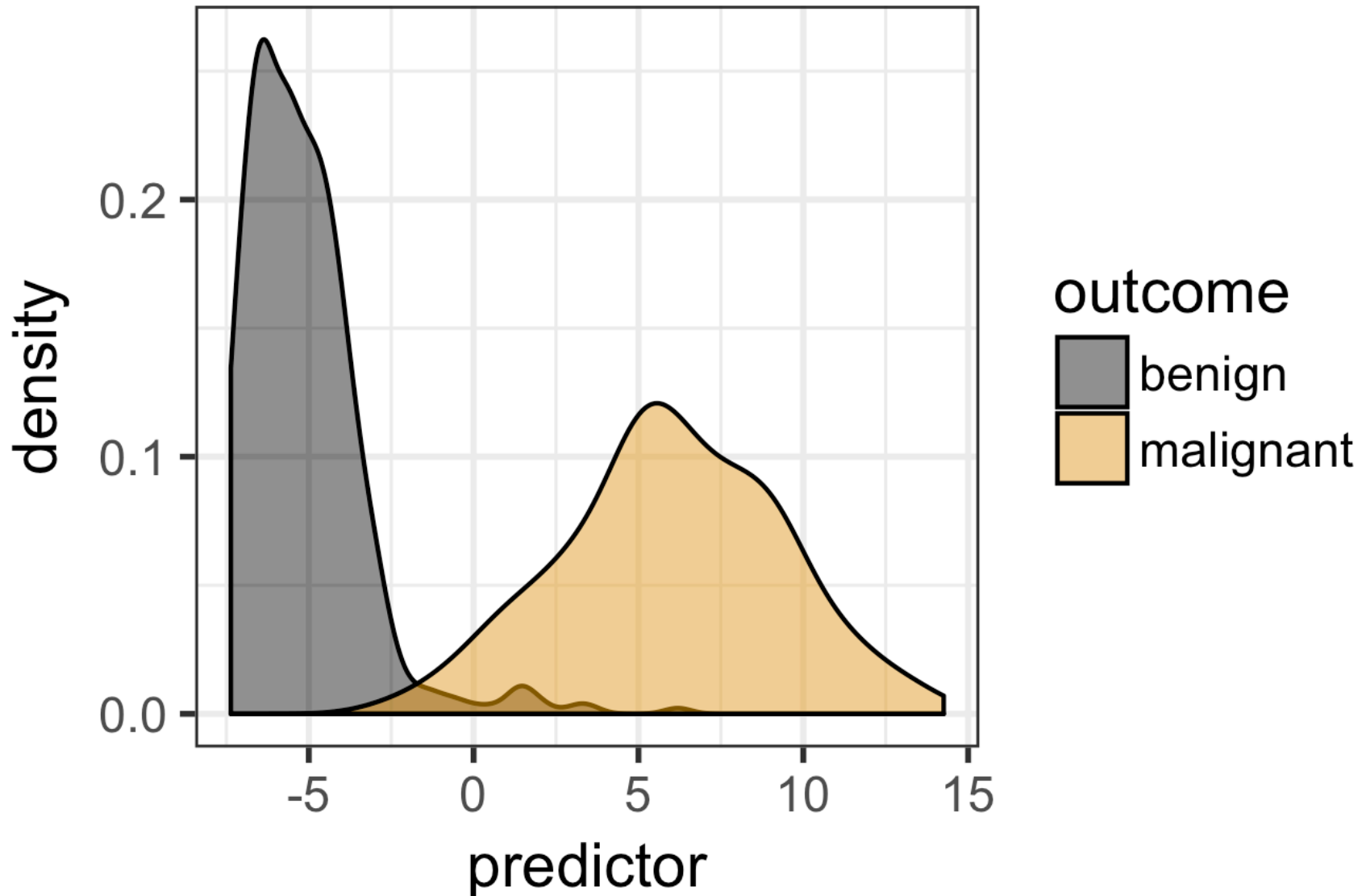
```
> glm.out$linear.predictors
```

	1	2	3	4	5	6
	-3.93164737	2.11714436	-4.79816093	1.73213613	-3.87911098	11.13827708
	7	8	9	10	11	12
	-3.01482307	-5.44973218	-6.74191480	-5.02551514	-6.42177489	-6.27057890
	13	14	15	16	17	18
	-0.59960855	-5.66467448	7.31555568	1.03125059	-5.02551514	-4.55417925
	19	20	21	22	23	24
	6.84403543	-3.30911549	5.77987063	7.59930618	-5.64804702	-6.42177489
	25	26	27	28	29	30
	1.26713222	-5.64804702	-4.40298326	-6.27057890	-6.66542501	-5.64804702
	31	32	33	34	35	36
	-5.79924301	6.08220228	-5.46170888	-5.29853619	-6.27057890	8.99139484
	37	38	39	40	41	42
	-1.42377192	5.38263613	0.70417516	3.02382523	5.95265328	-0.84593335
	43	44	45	46	47	48
	7.88442916	-6.89311078	3.13488983	-6.89311078	-3.87911098	4.10006298
	49	50	51	52	53	54
	1.29082051	-0.35486010	5.44017479	4.85067163	6.80192104	4.33368959
	55	56	57	58	59	60
	4.91966368	0.65666514	1.11699791	1.61527962	2.23350656	-6.51456058
	61	62	63	64	65	66
	4.84027081	0.06333410	-6.89311078	0.86675068	-4.55417925	2.94053974

The linear predictor clearly separates benign and malignant outcomes



The linear predictor clearly separates benign and malignant outcomes



Predicting outcome for new data with the `predict()` function

```
> patient1 <- data.frame(clump_thickness = 1,  
                          uniform_cell_size = 1,  
                          uniform_cell_shape = 1,  
                          marg_adhesion = 1,  
                          epithelial_cell_size = 4,  
                          bare_nuclei = 3,  
                          bland_chromatin = 1,  
                          normal_nucleoli = 1,  
                          mitoses = 1)
```

Predicting outcome for new data with the `predict()` function

```
> patient1 <- data.frame(clump_thickness = 1,  
                          uniform_cell_size = 1,  
                          uniform_cell_shape = 1,  
                          marg_adhesion = 1,  
                          epithelial_cell_size = 4,  
                          bare_nuclei = 3,  
                          bland_chromatin = 1,  
                          normal_nucleoli = 1,  
                          mitoses = 1)
```

```
> predict(glm.out, patient1) # linear predictor  
      1  
-6.607346
```

Predicting outcome for new data with the `predict()` function

```
> patient1 <- data.frame(clump_thickness = 1,  
                          uniform_cell_size = 1,  
                          uniform_cell_shape = 1,  
                          marg_adhesion = 1,  
                          epithelial_cell_size = 4,  
                          bare_nuclei = 3,  
                          bland_chromatin = 1,  
                          normal_nucleoli = 1,  
                          mitoses = 1)  
  
> predict(glm.out, patient1) # linear predictor  
      1  
-6.607346  
  
> predict(glm.out, patient1, type="response") # probability  
      1  
0.00134859
```

Predicting outcome for new data with the `predict()` function

```
> patient2 <- data.frame(clump_thickness = 4,  
                          uniform_cell_size = 5,  
                          uniform_cell_shape = 5,  
                          marg_adhesion = 10,  
                          epithelial_cell_size = 4,  
                          bare_nuclei = 10,  
                          bland_chromatin = 7,  
                          normal_nucleoli = 5,  
                          mitoses = 8)
```

Predicting outcome for new data with the `predict()` function

```
> patient2 <- data.frame(clump_thickness = 4,  
                          uniform_cell_size = 5,  
                          uniform_cell_shape = 5,  
                          marg_adhesion = 10,  
                          epithelial_cell_size = 4,  
                          bare_nuclei = 10,  
                          bland_chromatin = 7,  
                          normal_nucleoli = 5,  
                          mitoses = 8)
```

```
> predict(glm.out, patient2) # linear predictor
```

1

6.14665

Predicting outcome for new data with the `predict()` function

```
> patient2 <- data.frame(clump_thickness = 4,  
                          uniform_cell_size = 5,  
                          uniform_cell_shape = 5,  
                          marg_adhesion = 10,  
                          epithelial_cell_size = 4,  
                          bare_nuclei = 10,  
                          bland_chromatin = 7,  
                          normal_nucleoli = 5,  
                          mitoses = 8)  
  
> predict(glm.out, patient2) # linear predictor  
      1  
6.14665  
  
> predict(glm.out, patient2, type="response") # probability  
      1  
0.9978639
```