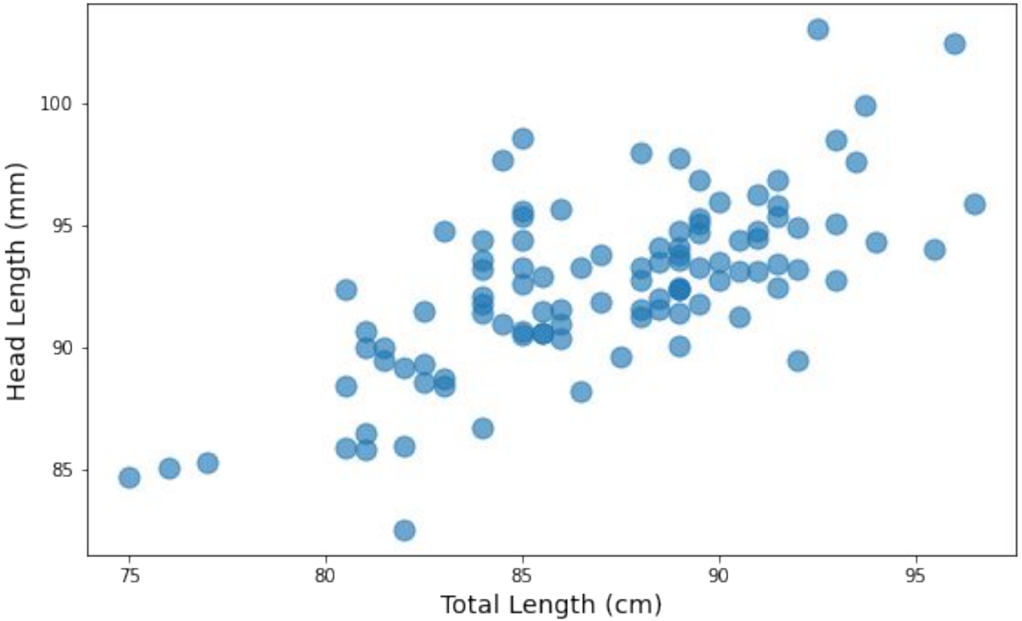
Introduction to Generalized Linear Models

Part 1: Linear Regression



Goal: Predict an Australian brushtail possum's head length

OpenIntro Statistics, Section 8.1.2

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
possum.head()
```

	site	pop	sex	age	head_l	skull_w	total_l	tail_l
0	1	Vic	m	8.0	94.1	60.4	89.0	36.0
1	1	Vic	f	6.0	92.5	57.6	91.5	36.5
2	1	Vic	f	6.0	94.0	60.0	95.5	39.0
3	1	Vic	f	6.0	93.2	57.1	92.0	38.0
4	1	Vic	f	2.0	91.5	56.3	85.5	36.0

target column

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
import statsmodels.formula.api as smf

linreg = smf.ols('head_l ~ 1', data = possum).fit()
```

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
import statsmodels.formula.api as smf

linreg = smf.ols('head_l ~ 1', data = possum).fit()
```

We'll be using the *statsmodels* library and the formula api.

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
import statsmodels.formula.api as smf

linreg = smf.ols('head_l ~ 1', data = possum).fit()
```

Create an ordinary least squares (ols) model.

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
import statsmodels.formula.api as smf

linreg = smf.ols('head_l ~ 1', data = possum).fit()
```

Give a patsy formula for out model.

'target ~ predictor(s)'

Using just 1 as a predictor will fit only a constant.

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
import statsmodels.formula.api as smf

linreg = smf.ols('head_l ~ 1', data = possum).fit()
```

DataFrame containing the data.

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
import statsmodels.formula.api as smf

linreg = smf.ols('head_l ~ 1', data = possum) .fit()
```

Go ahead and fit the model
after specifying it.

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
linreg.summary()
```

OLS Regression Results

Dep. Variable:	head_l	**R-squared:**	-0.000
Model:	OLS	**Adj. R-squared:**	-0.000
Method:	Least Squares	**F-statistic:**	nan
Date:	Wed, 28 Sep 2022	**Prob (F-statistic):**	nan
Time:	22:00:48	**Log-Likelihood:**	-279.51
No. Observations:	104	**AIC:**	561.0
Df Residuals:	103	**BIC:**	563.7
Df Model:	0		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	92.6029	0.350	264.281	0.000	91.908	93.298

Omnibus:	2.844	**Durbin-Watson:**	1.203
Prob(Omnibus):	0.241	**Jarque-Bera (JB):**	2.767
Skew:	-0.055	**Prob(JB):**	0.251
Kurtosis:	3.791	**Cond. No.**	1.00

Approach 1: Ignore the Total Length variable, and just look at the overall distribution of Head Length.

```
linreg.summary()
```

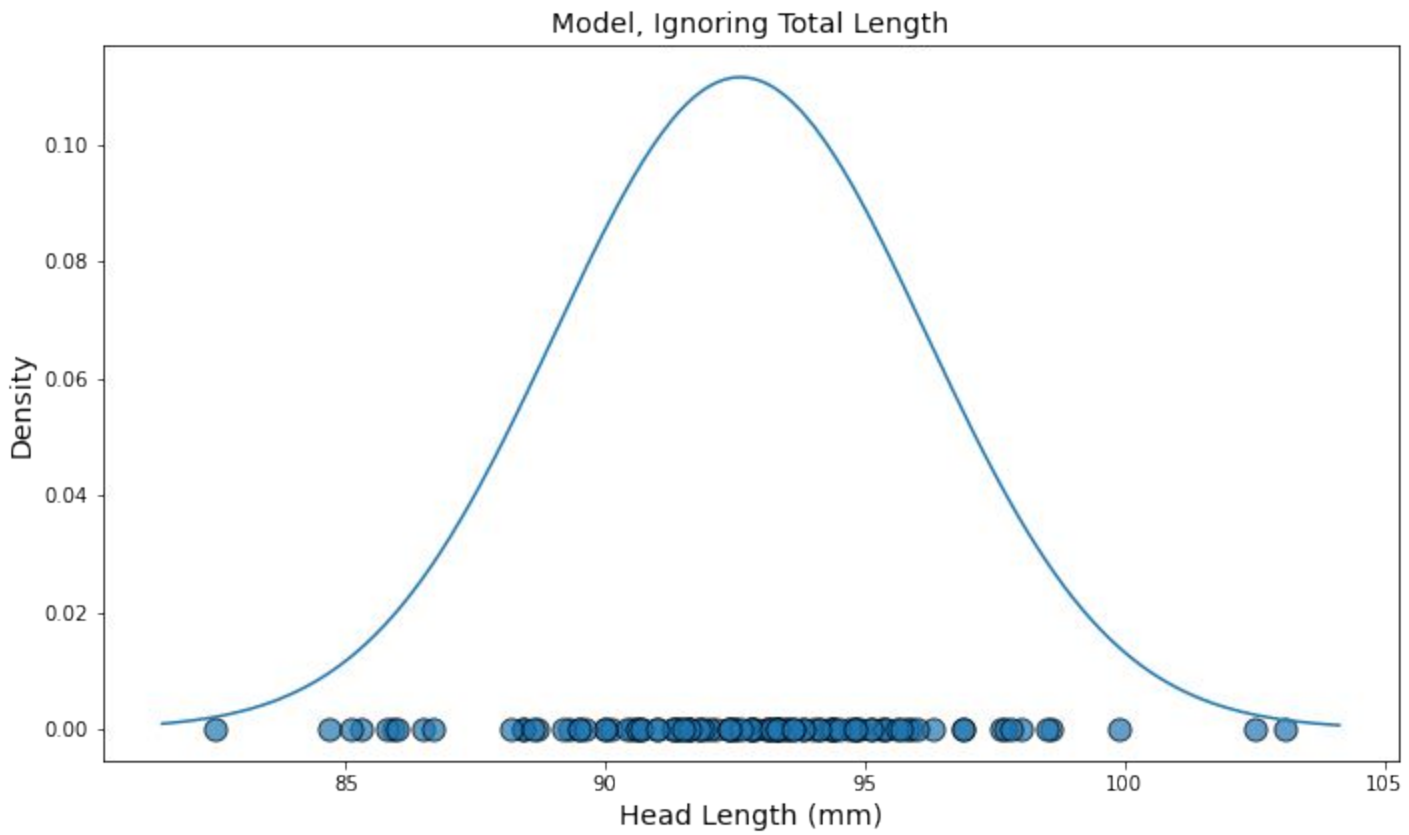
OLS Regression Results

Dep. Variable:	head_l	**R-squared:**	-0.000
Model:	OLS	**Adj. R-squared:**	-0.000
Method:	Least Squares	**F-statistic:**	nan
Date:	Wed, 28 Sep 2022	**Prob (F-statistic):**	nan
Time:	22:00:48	**Log-Likelihood:**	-279.51
No. Observations:	104	**AIC:**	561.0
Df Residuals:	103	**BIC:**	563.7
Df Model:	0		
Covariance Type:	nonrobust		

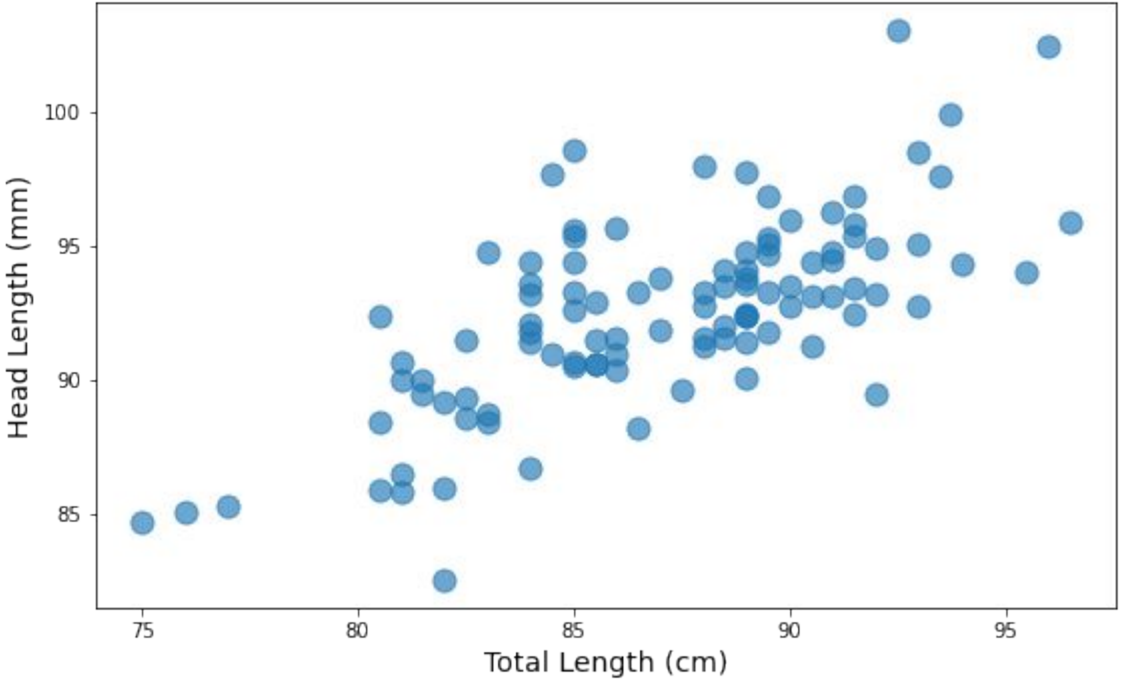
	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	92.6029	0.350	264.281	0.000	91.908	93.298

Omnibus:	2.844	**Durbin-Watson:**	1.203
Prob(Omnibus):	0.241	**Jarque-Bera (JB):**	2.767
Skew:	-0.055	**Prob(JB):**	0.251
Kurtosis:	3.791	**Cond. No.**	1.00

The estimated mean of the distribution of head lengths is 92.6029.



Model, Ignoring Total Length
Density
Head Length (mm)
0.00
0.02
0.04
0.06
0.08
0.10
85
90
95
100
105



The results from approach 1 look *okay*, but we are disregarding a lot of potentially useful information - the total length measurement.

Approach 2: Predict using the total length (and a constant).

Approach 2: Predict using the total length (and a constant).

```
linreg_tl = smf.ols('head_l ~ total_l', data = possum).fit()
```

Approach 2: Predict using the total length (and a constant).

```
linreg_tl = smf.ols('head_l ~  total_l', data = possum).fit()
```

This time, we'll use the total length column as a predictor.

Approach 2: Predict using the total length (and a constant).

```
linreg_tl.summary()
```

OLS Regression Results

Dep. Variable:	head_l	**R-squared:**	0.478
Model:	OLS	**Adj. R-squared:**	0.472
Method:	Least Squares	**F-statistic:**	93.26
Date:	Wed, 28 Sep 2022	**Prob (F-statistic):**	4.68e-16
Time:	21:55:29	**Log-Likelihood:**	-245.75
No. Observations:	104	**AIC:**	495.5
Df Residuals:	102	**BIC:**	500.8
Df Model:	1		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	42.7098	5.173	8.257	0.000	32.450	52.970
total_l	0.5729	0.059	9.657	0.000	0.455	0.691

Omnibus:	5.577	**Durbin-Watson:**	1.881
Prob(Omnibus):	0.062	**Jarque-Bera (JB):**	5.117
Skew:	0.422	**Prob(JB):**	0.0774
Kurtosis:	3.684	**Cond. No.**	1.77e+03

For possums with a total length of t, the model estimates that the distribution of head lengths is normal with a mean of $42.7098 + 0.5729t$.

Approach 2: Predict using the total length (and a constant).

```
linreg_tl.summary()
```

OLS Regression Results

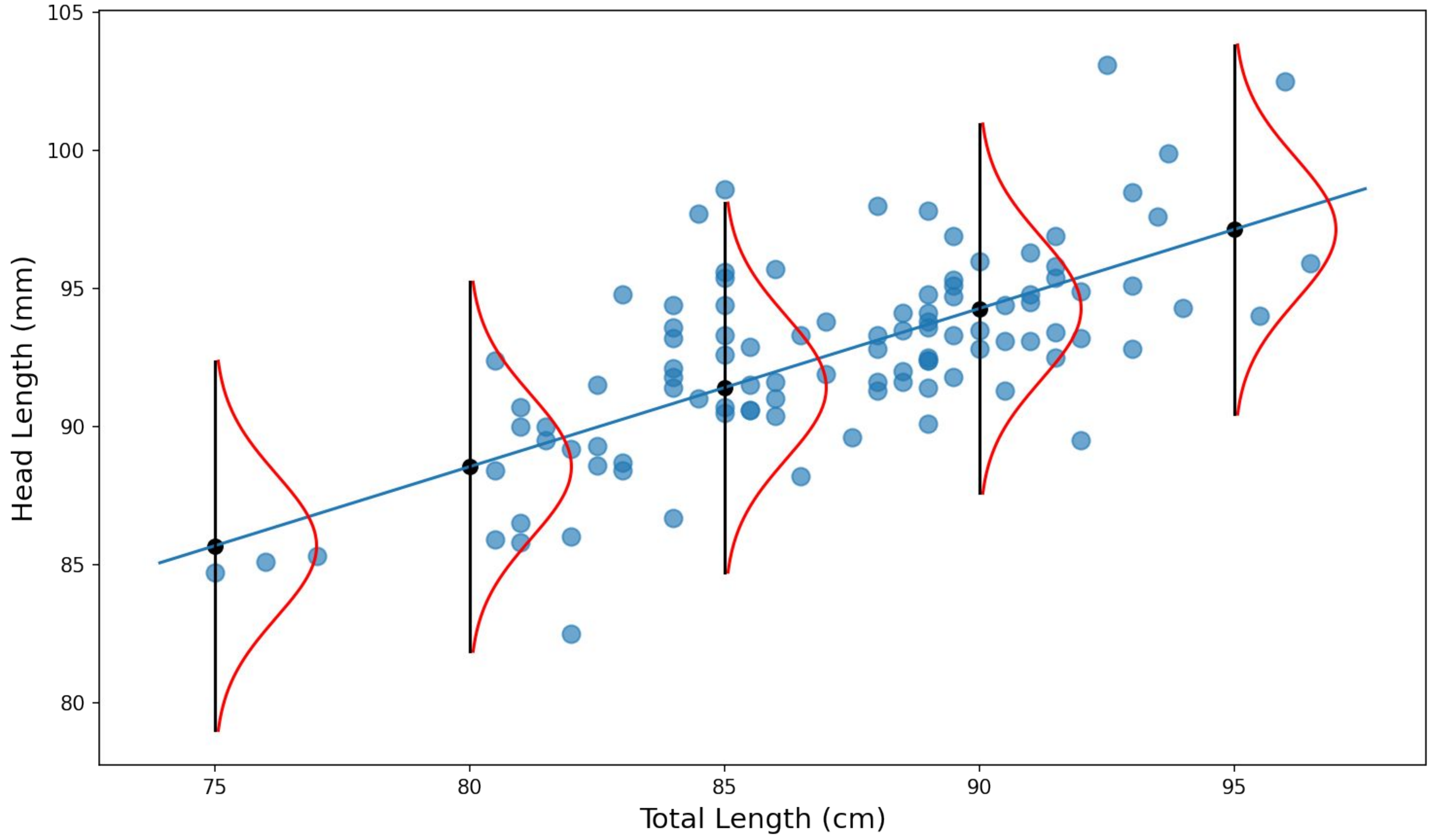
Dep. Variable:	head_l	**R-squared:**	0.478
Model:	OLS	**Adj. R-squared:**	0.472
Method:	Least Squares	**F-statistic:**	93.26
Date:	Wed, 28 Sep 2022	**Prob (F-statistic):**	4.68e-16
Time:	21:55:29	**Log-Likelihood:**	-245.75
No. Observations:	104	**AIC:**	495.5
Df Residuals:	102	**BIC:**	500.8
Df Model:	1		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	42.7098	5.173	8.257	0.000	32.450	52.970
total_l	0.5729	0.059	9.657	0.000	0.455	0.691

Omnibus:	5.577	**Durbin-Watson:**	1.881
Prob(Omnibus):	0.062	**Jarque-Bera (JB):**	5.117
Skew:	0.422	**Prob(JB):**	0.0774
Kurtosis:	3.684	**Cond. No.**	1.77e+03

For possums with a total length of t, the model estimates that the distribution of head lengths is normal with a mean of $42.7098 + 0.5729t$.

A one-unit increase in total length is corresponds to an increase of 0.5729 in the estimated average head length.



Head Length (mm)
Total Length (cm)
105
100
95
90
85
80
75
80
85
90
95

Linear Regression

We have estimated the distribution of head lengths, *conditional* on the total length.

Linear Regression

We have estimated the distribution of head lengths, *conditional* on the total length.

If we let Y be the head length and x be the total length, we have estimated the distribution of $Y|x$.

Linear Regression

We have estimated the distribution of head lengths, *conditional* on the total length.

If we let Y be the head length and x be the total length, we have estimated the distribution of $Y|x$.

Specifically, we have said that it follows a normal distribution with mean $42.7098 + 0.5729x$

Linear Regression in General

$Y|x$ follows a normal distribution with mean

$$\mu = \beta_0 + \beta_1 x$$

Linear Regression

What if we have more predictors?

Linear Regression

What if we have more predictors?

For example, along with total length x_1, we could include skull width as x_2.

Approach 2: Predict using the total length (and a constant).

```
linreg_tlsw = smf.ols(
    'head_l ~ total_l + skull_w',
    data = possum
).fit()
```

We can just add this new predictor to our formula.

Approach 2: Predict using the total length (and a constant).

```
linreg_tlsw.summary()
```

OLS Regression Results

Dep. Variable:	head_l	**R-squared:**	0.644
Model:	OLS	**Adj. R-squared:**	0.637
Method:	Least Squares	**F-statistic:**	91.43
Date:	Thu, 29 Sep 2022	**Prob (F-statistic):**	2.17e-23
Time:	13:38:14	**Log-Likelihood:**	-225.78
No. Observations:	104	**AIC:**	457.6
Df Residuals:	101	**BIC:**	465.5
Df Model:	2		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	29.6127	4.694	6.309	0.000	20.301	38.924
total_l	0.3634	0.058	6.280	0.000	0.249	0.478
skull_w	0.5510	0.080	6.877	0.000	0.392	0.710

From the output, we've estimated that Y|(x1, x2) is normal with mean

$$\mu = 29.62 + 0.36x_1 + 0.55x_2$$

Approach 2: Predict using the total length (and a constant).

```
linreg_tlsw.summary()
```

OLS Regression Results

Dep. Variable:	head_l	**R-squared:**	0.644
Model:	OLS	**Adj. R-squared:**	0.637
Method:	Least Squares	**F-statistic:**	91.43
Date:	Thu, 29 Sep 2022	**Prob (F-statistic):**	2.17e-23
Time:	13:38:14	**Log-Likelihood:**	-225.78
No. Observations:	104	**AIC:**	457.6
Df Residuals:	101	**BIC:**	465.5
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	29.6127	4.694	6.309	0.000	20.301	38.924
total_l	0.3634	0.058	6.280	0.000	0.249	0.478
skull_w	0.5510	0.080	6.877	0.000	0.392	0.710

From the output, we've estimated that Y|(x1, x2) is normal with mean

$$\mu = 29.62 + 0.36x_1 + 0.55x_2$$

A one-unit increase in total length leads to a 0.36 unit increase in the estimated mean head length, **holding skull length constant**.

Linear Regression in General

$Y|\vec{x}$ follows a normal distribution with mean

$$\mu = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$$

Where $\vec{x} = \langle x_1, \ldots, x_n \rangle$ are the values of the predictor variables.