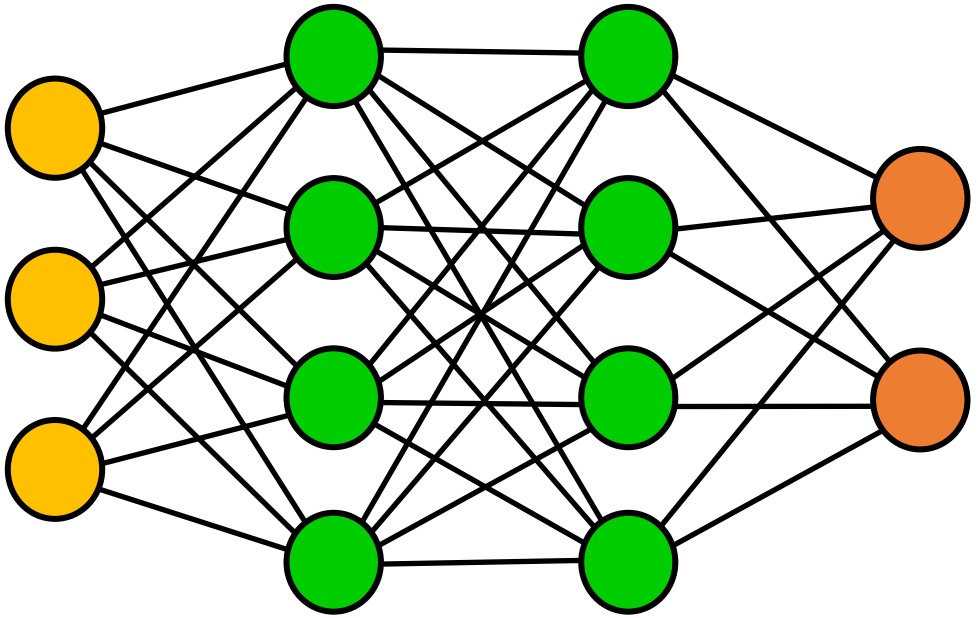
ANN vs Regression

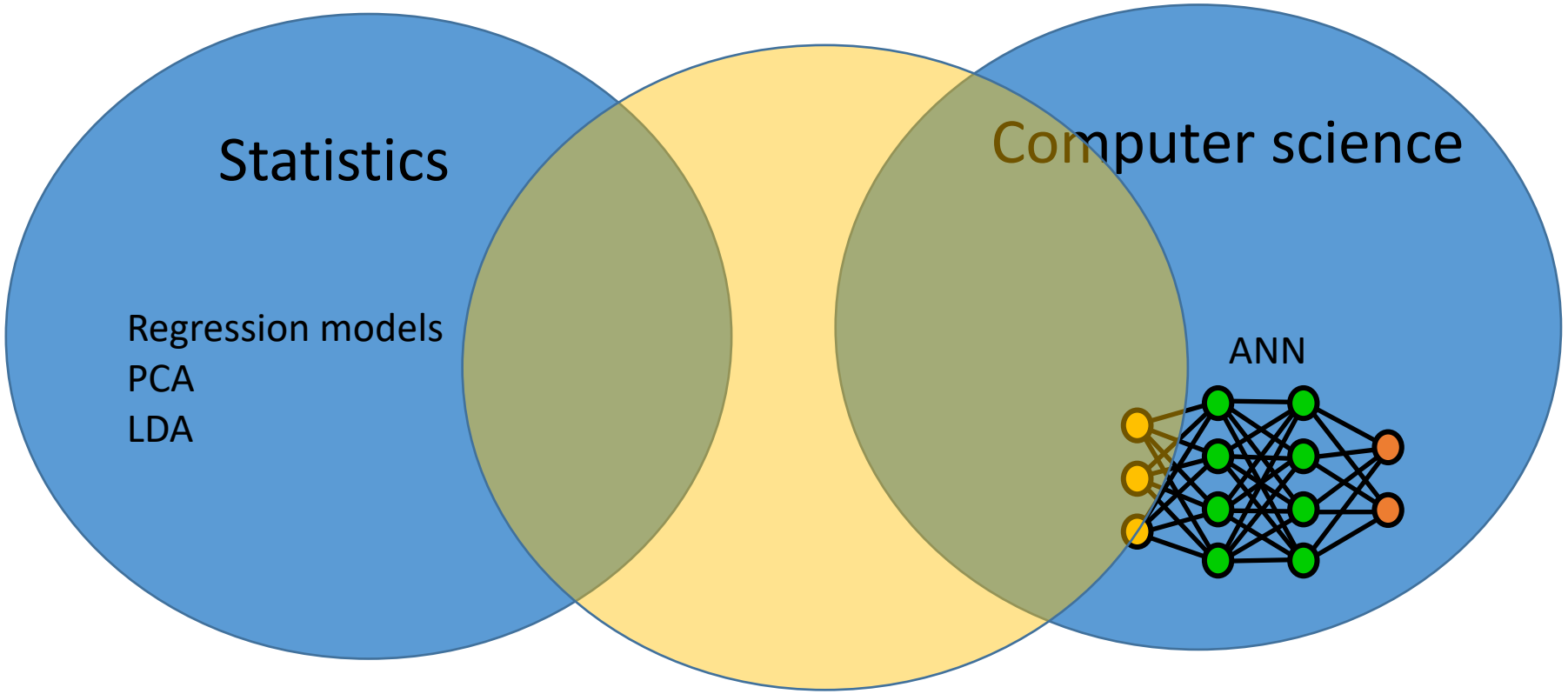
Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8



HÖGSKOLAN I SKÖVDE

Andreas Tilevik
University of Skövde

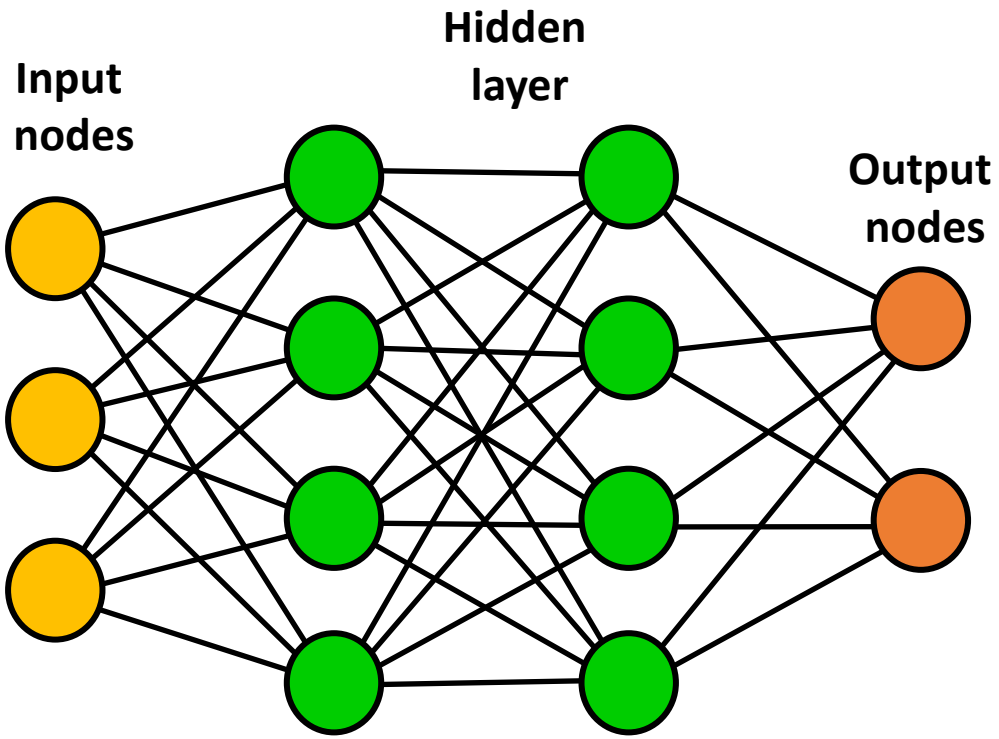
ANN vs Regression



Note!

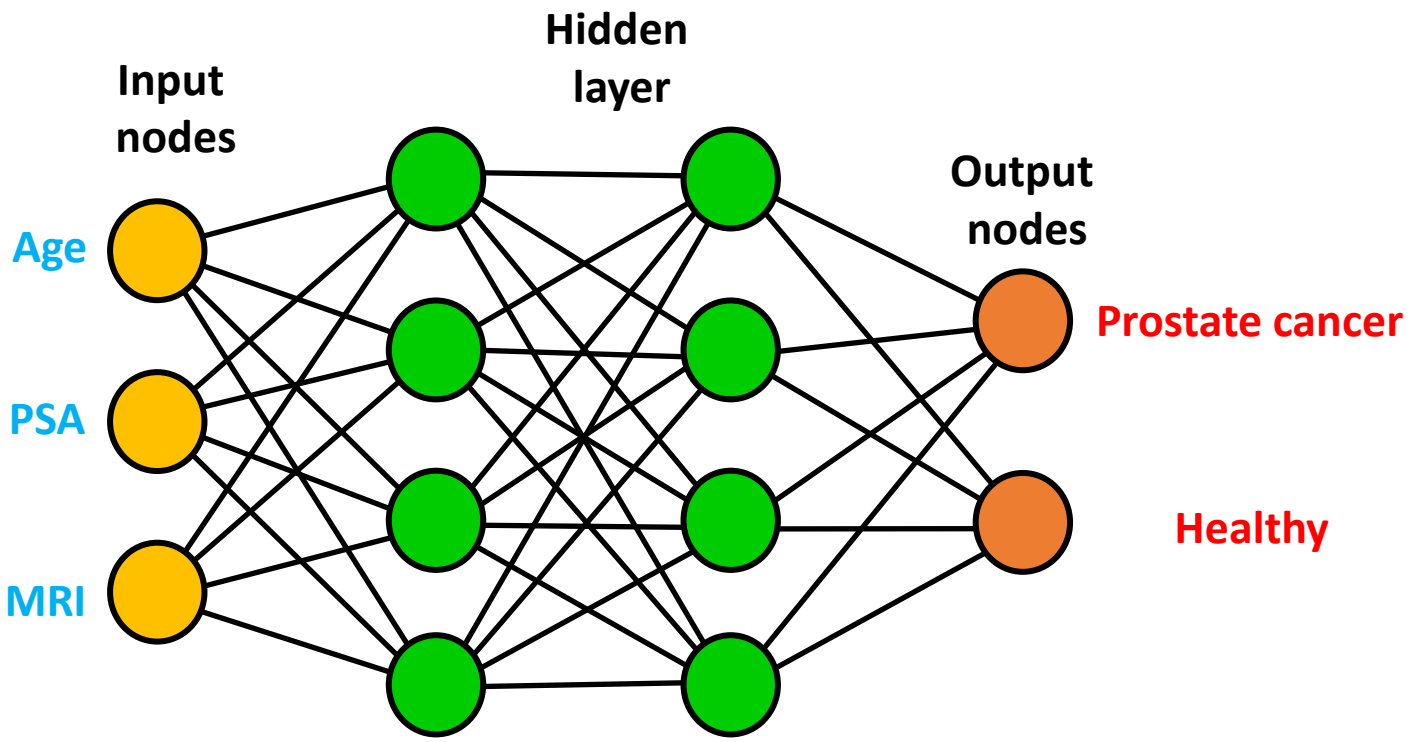
To simplify the explanations in this lecture, I have not normalized the data. If you will use neural networks in the future, make sure that you always normalize the data before training a network.

ANN – the basics

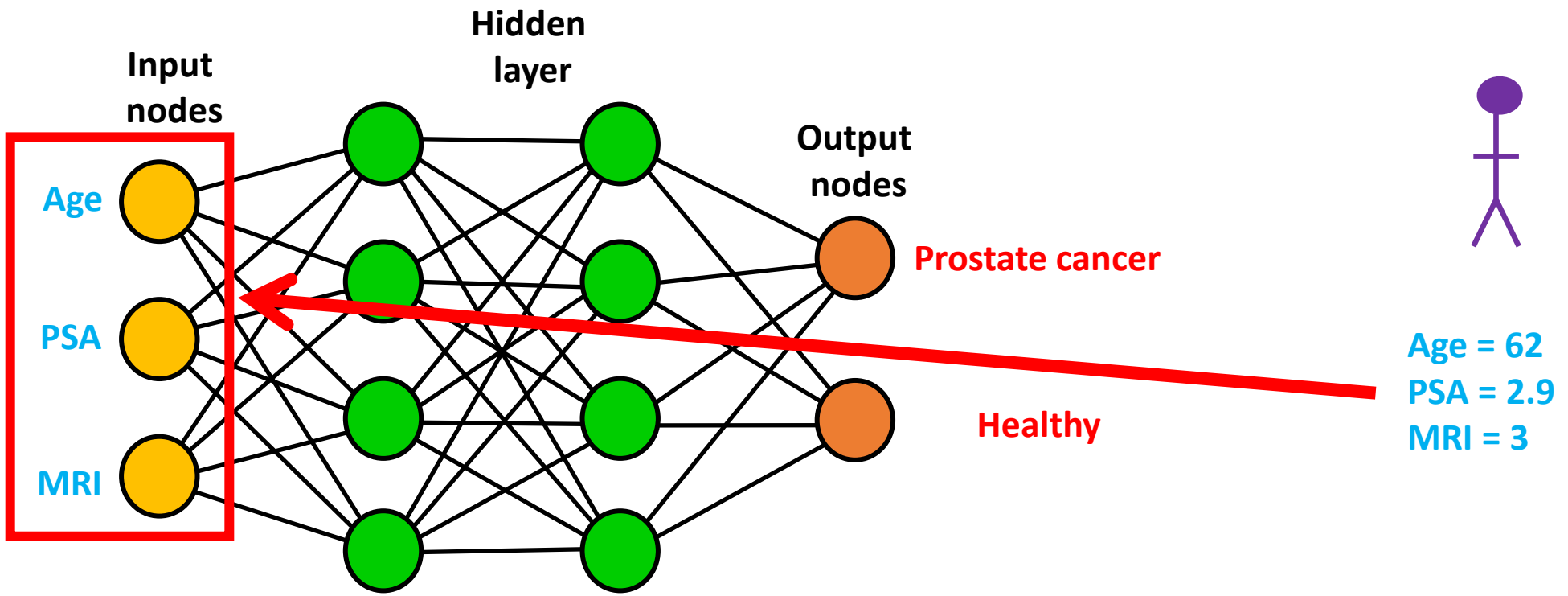


Deep learning – the use of a neural network with more than one hidden layer.

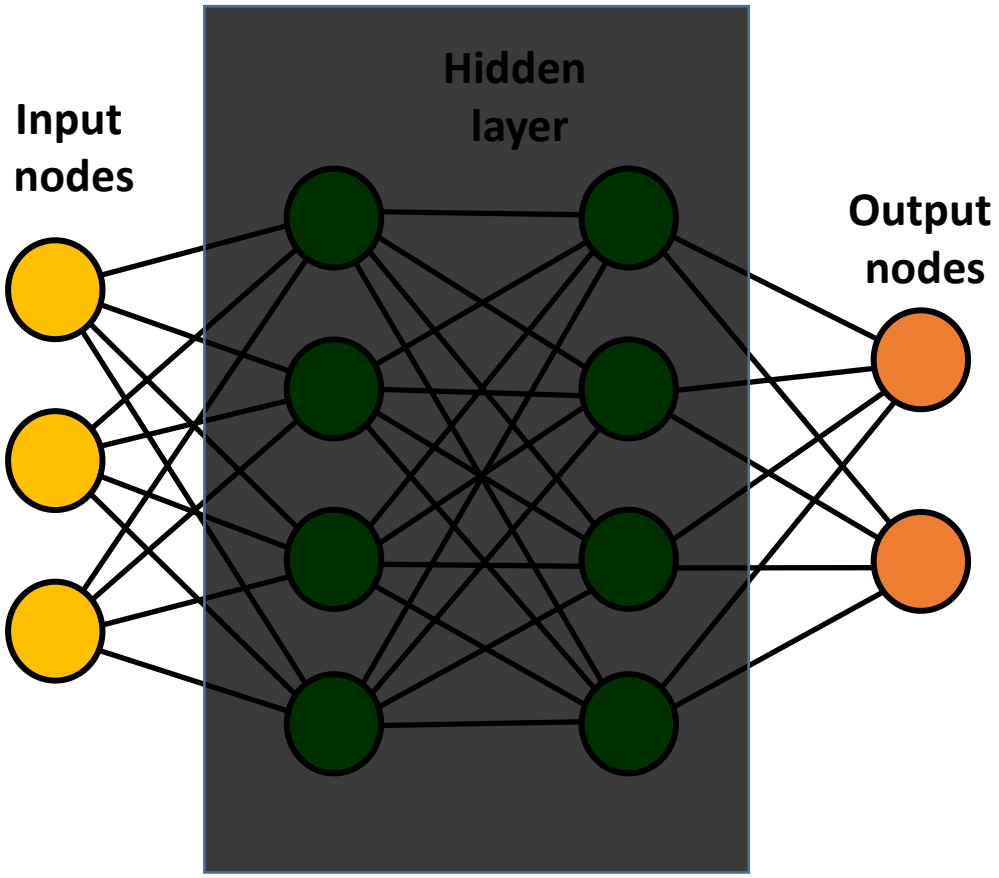
ANN – the basics



ANN – the basics



ANN – the basics



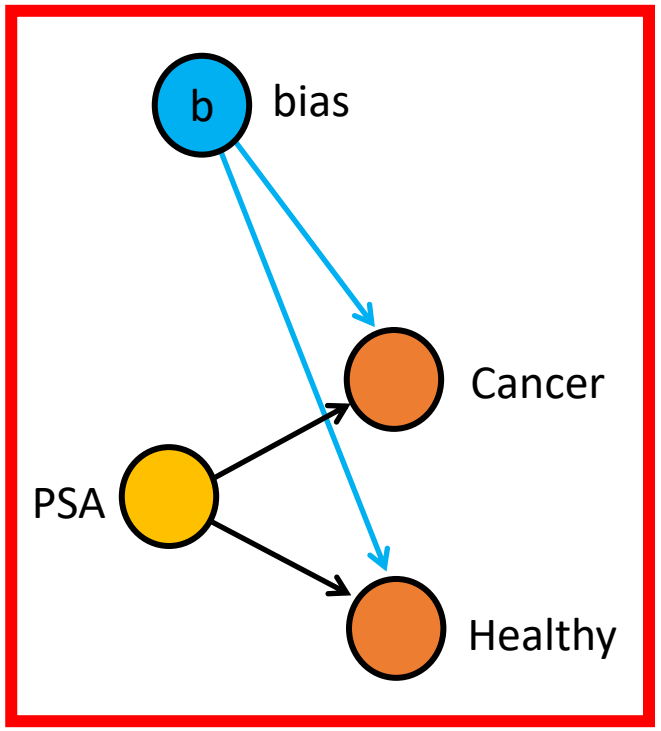
Black box

The data

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8

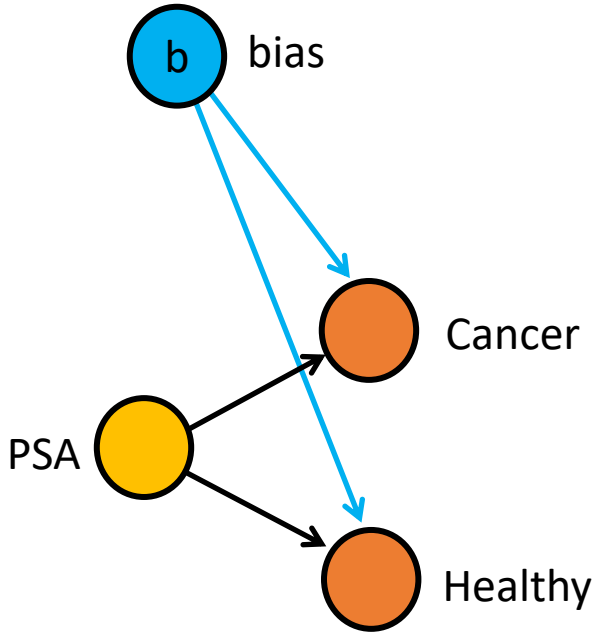
The simplest possible neural network

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8



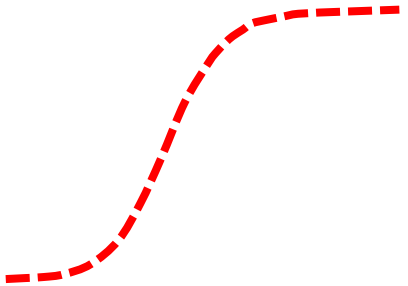
Select an activation function

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8



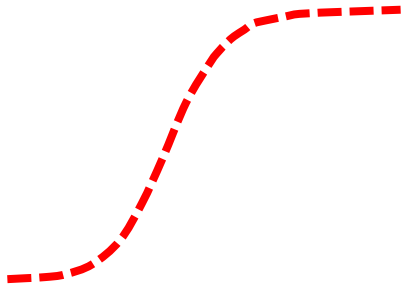
Sigmoid

$$f(z) = \frac{1}{1 + e^{-z}}$$



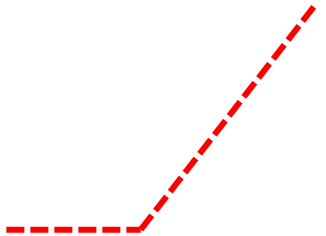
Tanh

$$f(z) = \tanh(z)$$



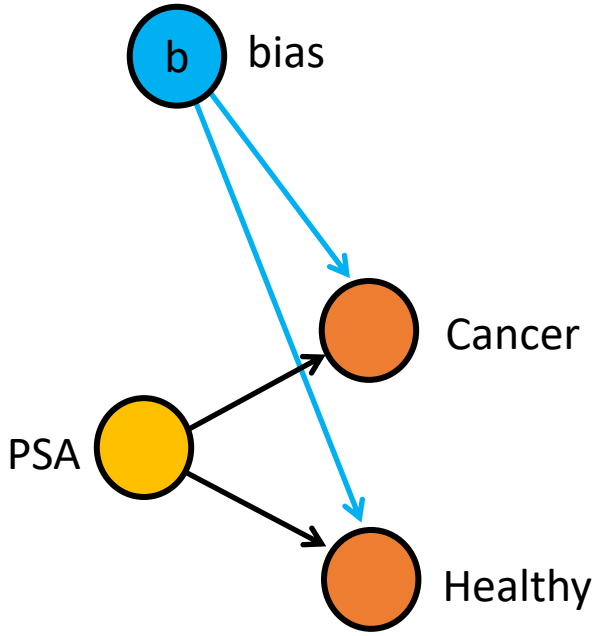
ReLU

$$f(z) = \max(0, z)$$



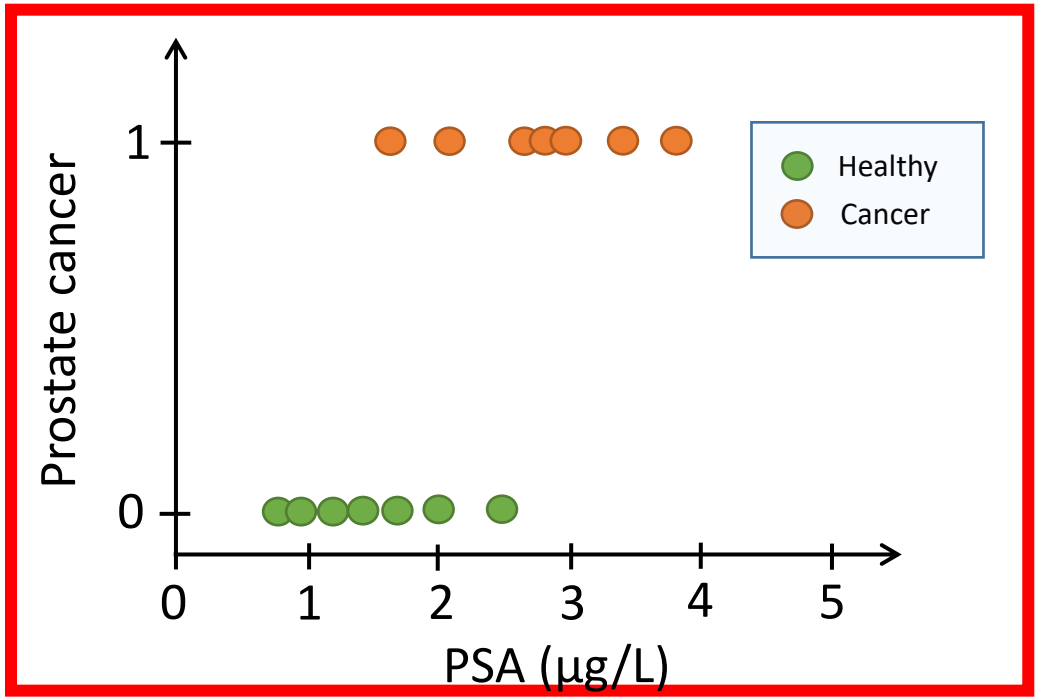
Plot the data

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8



Activation function

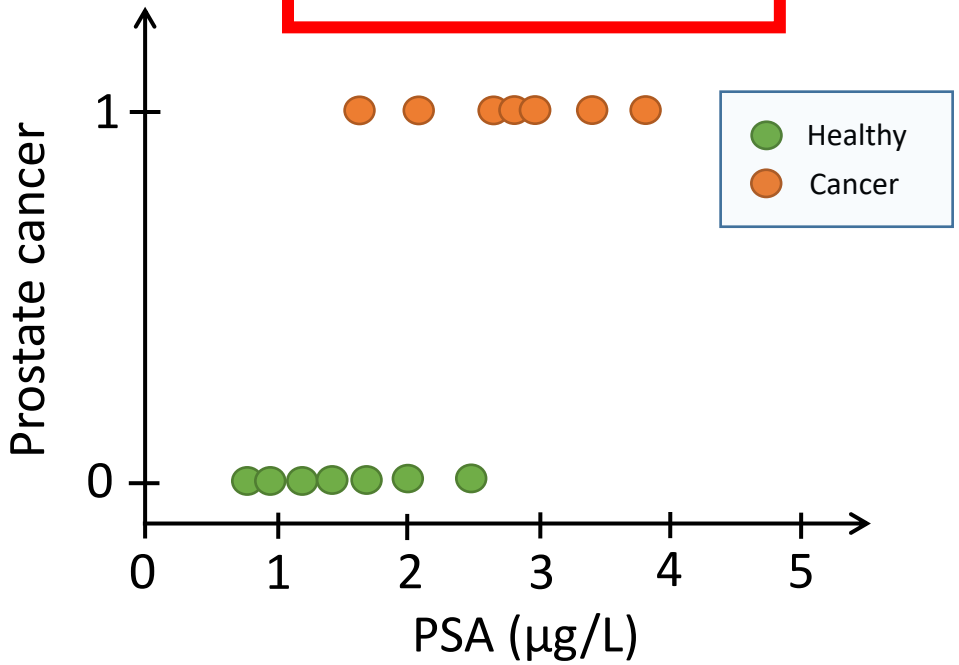
$$f(z) = \frac{1}{1 + e^{-z}}$$

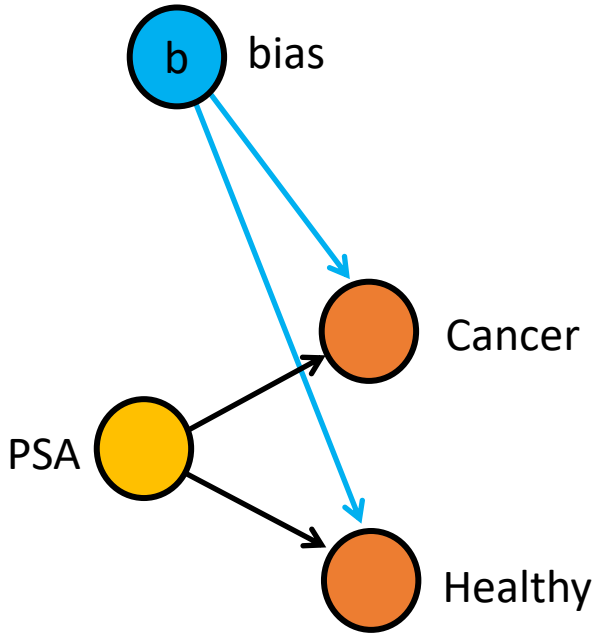


The math

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

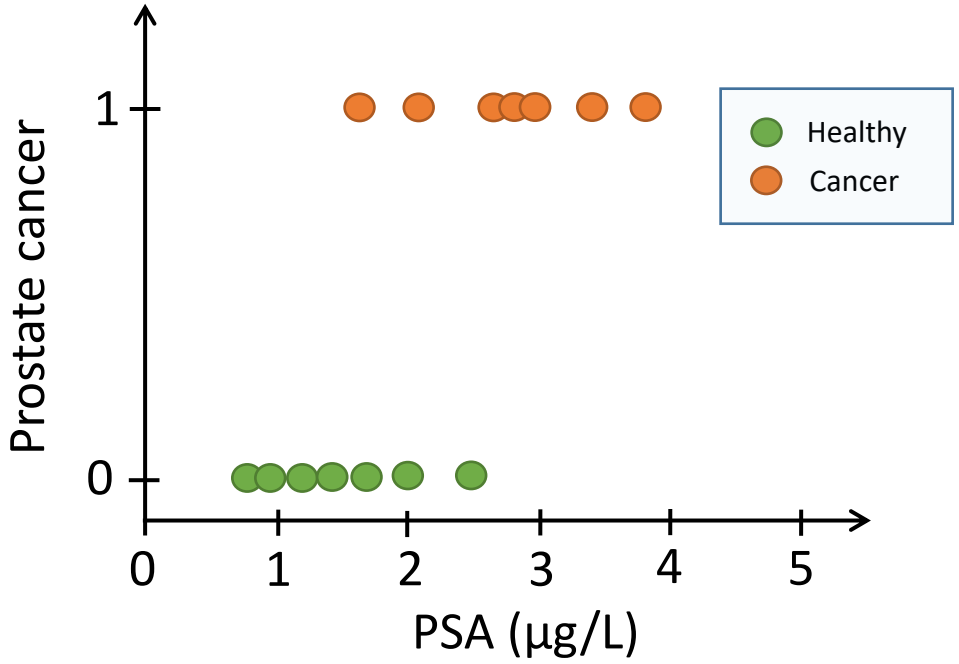


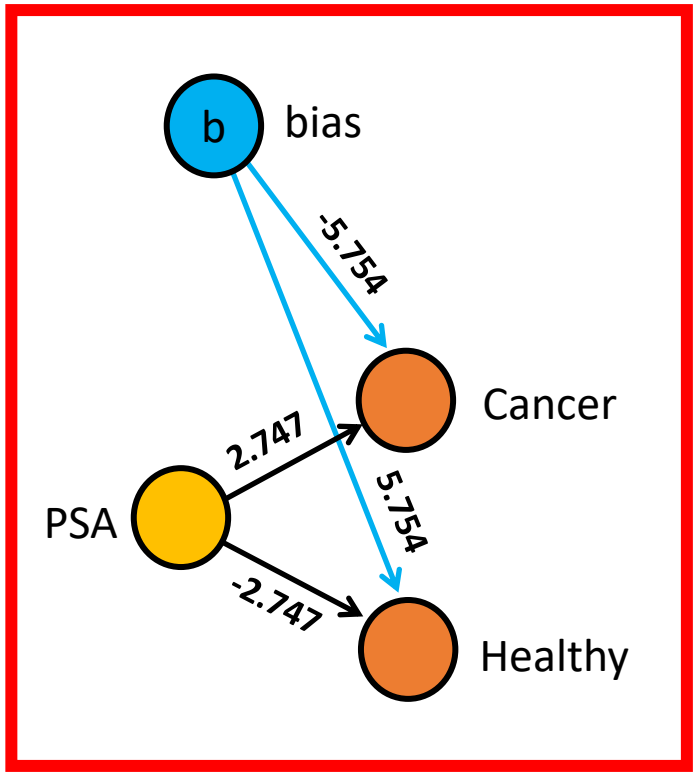


The math

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

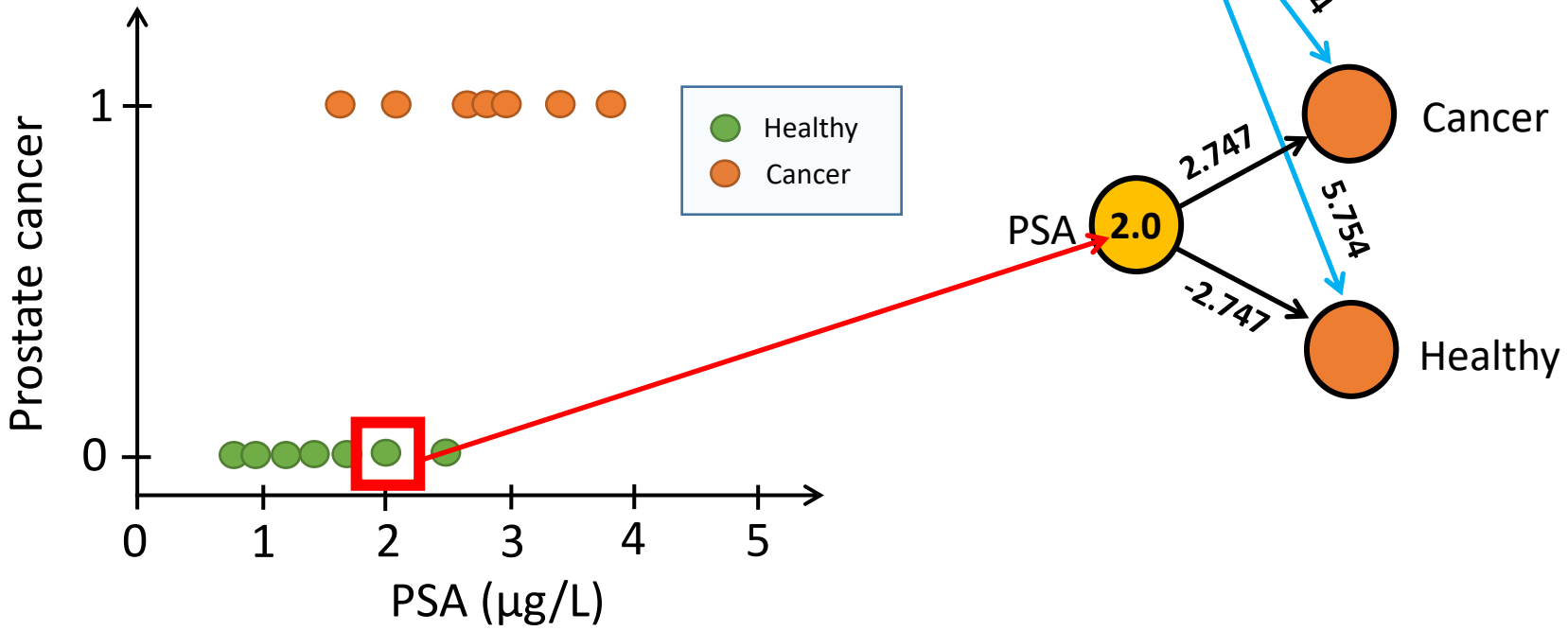




Classification

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

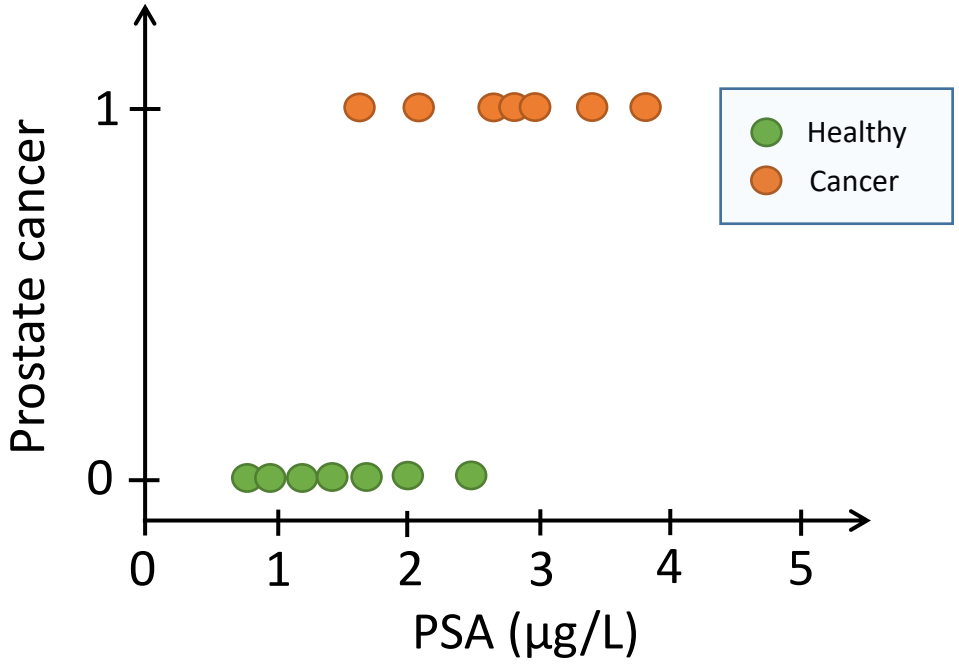
$$f(z) = \frac{1}{1 + e^{-z}}$$



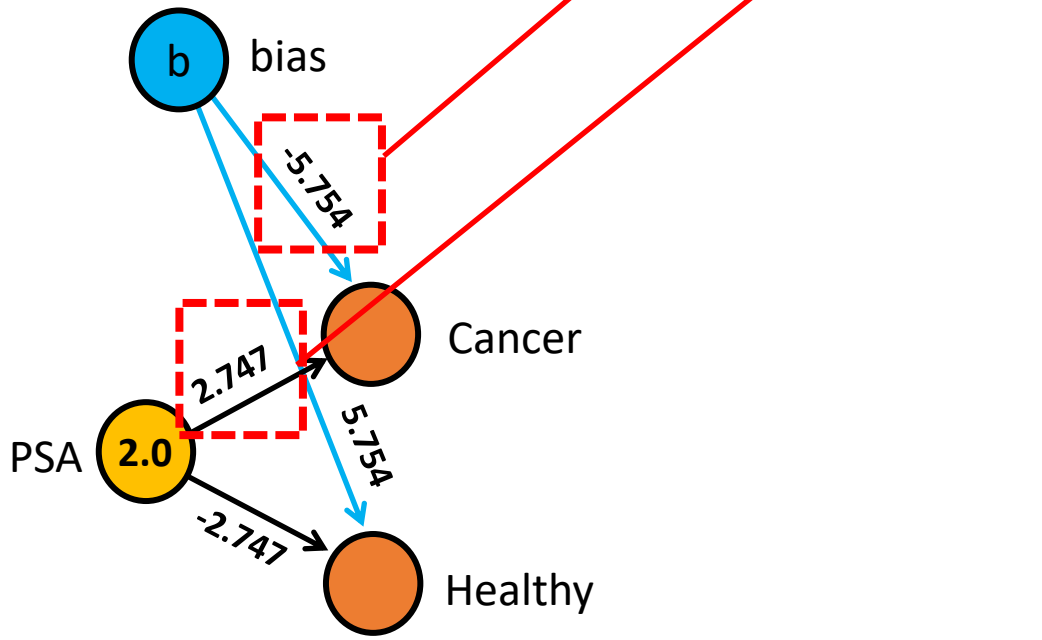
Classification

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



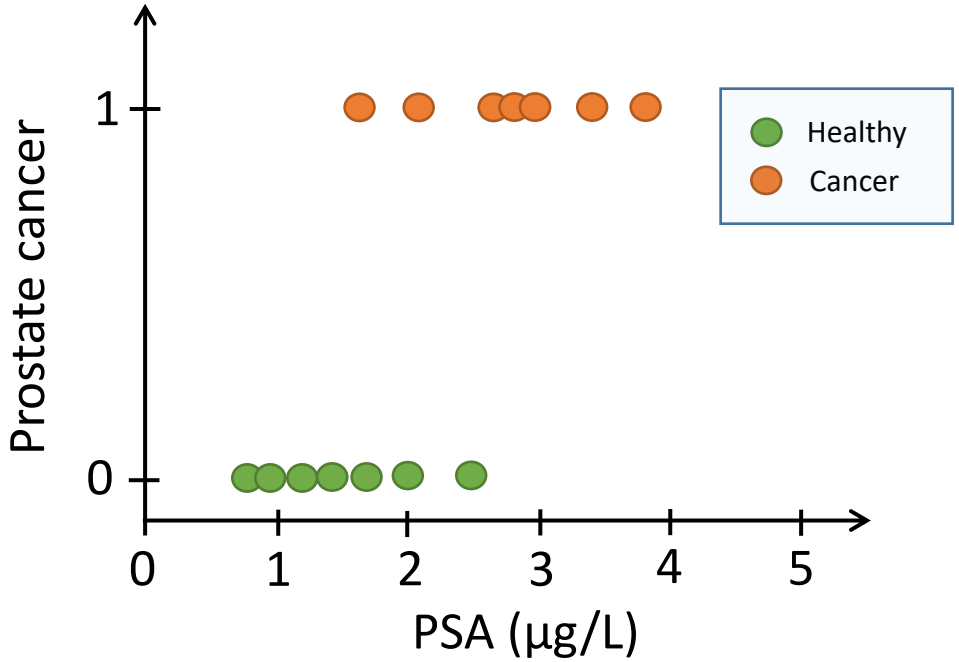
$$z = -5.754 + 2.747 \cdot 2.0$$



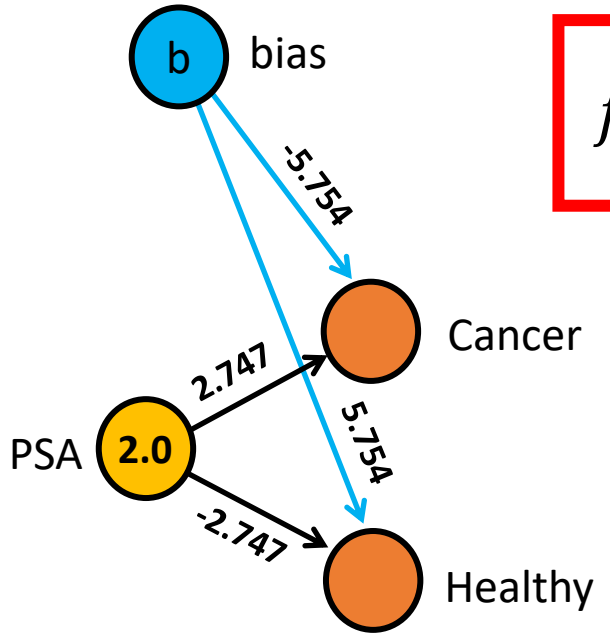
Classification

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



$$z = -5.754 + 2.747 \cdot 2.0 = -0.251$$

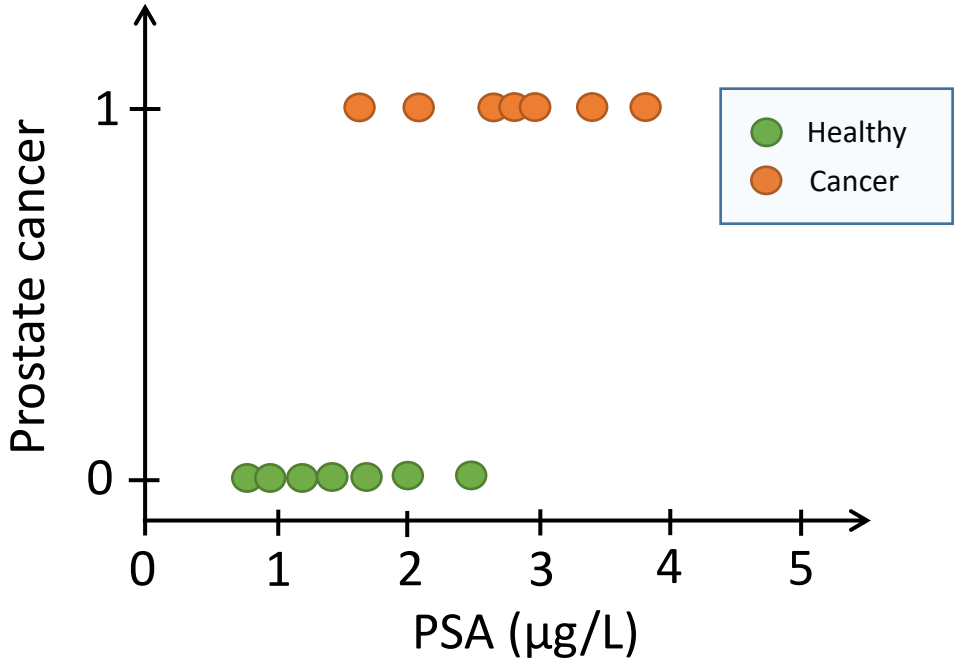


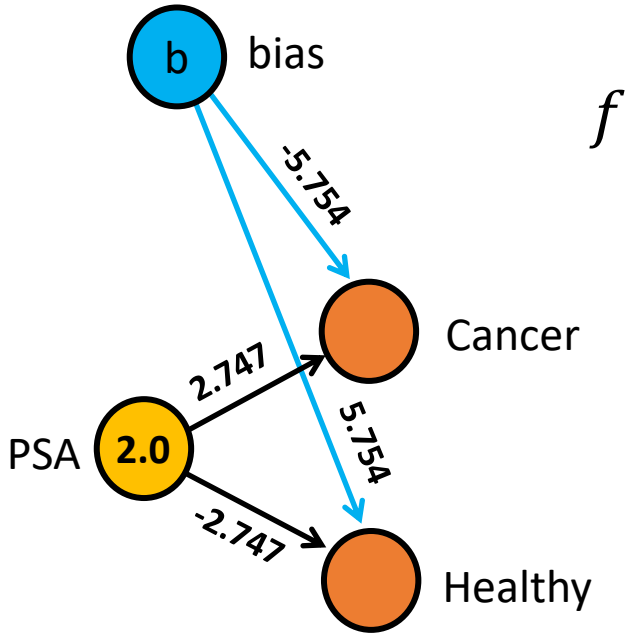
$$f(z) = \frac{1}{1 + e^{-z}}$$

Classification

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$





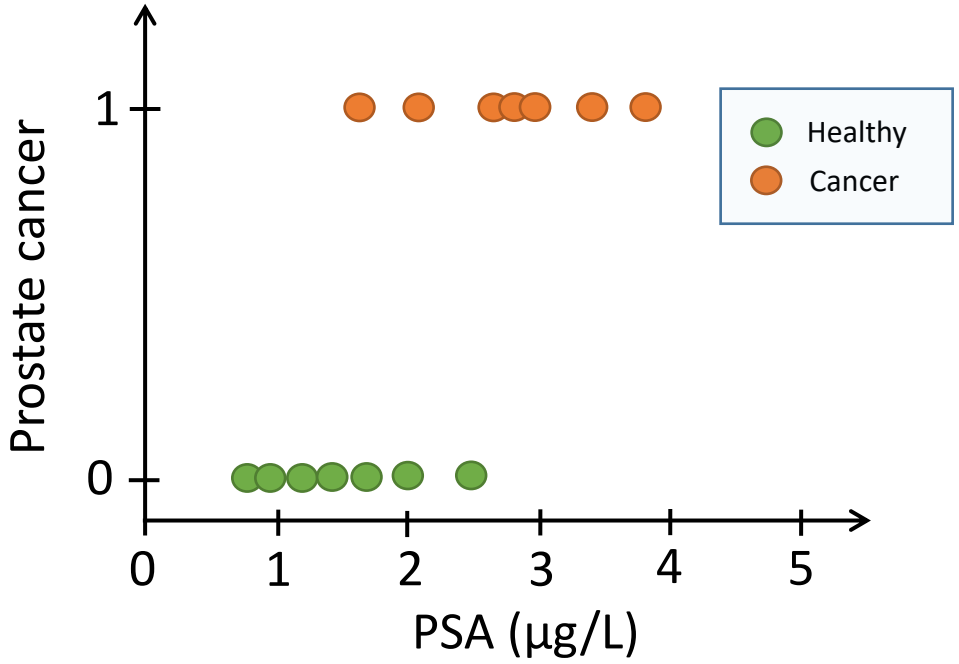
$$z = -5.754 + 2.747 \cdot 2.0 = -0.251$$

$$f(z) = \frac{1}{1 + e^{-(-0.251)}} = 0.438$$

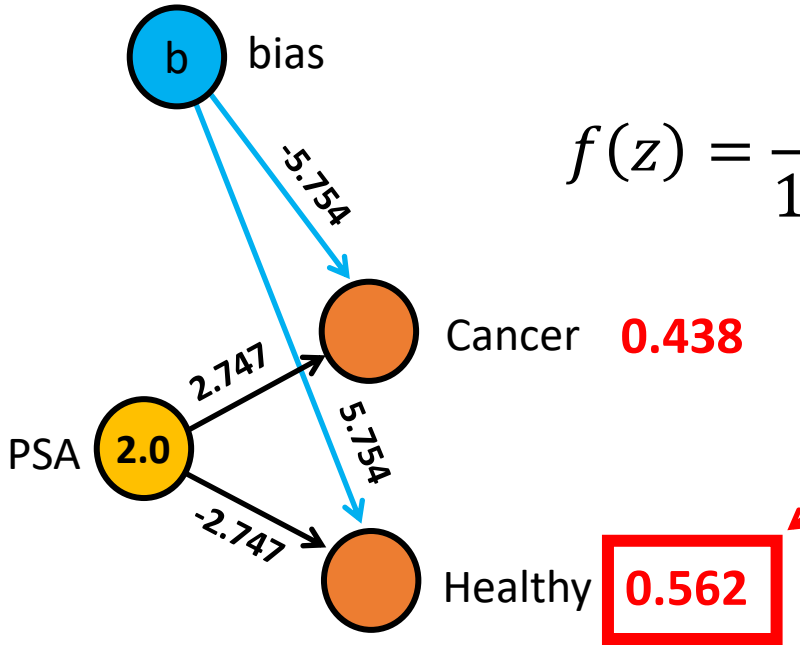
Classification

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



$$z = 5.754 + (-2.747) \cdot 2.0 = 0.251$$

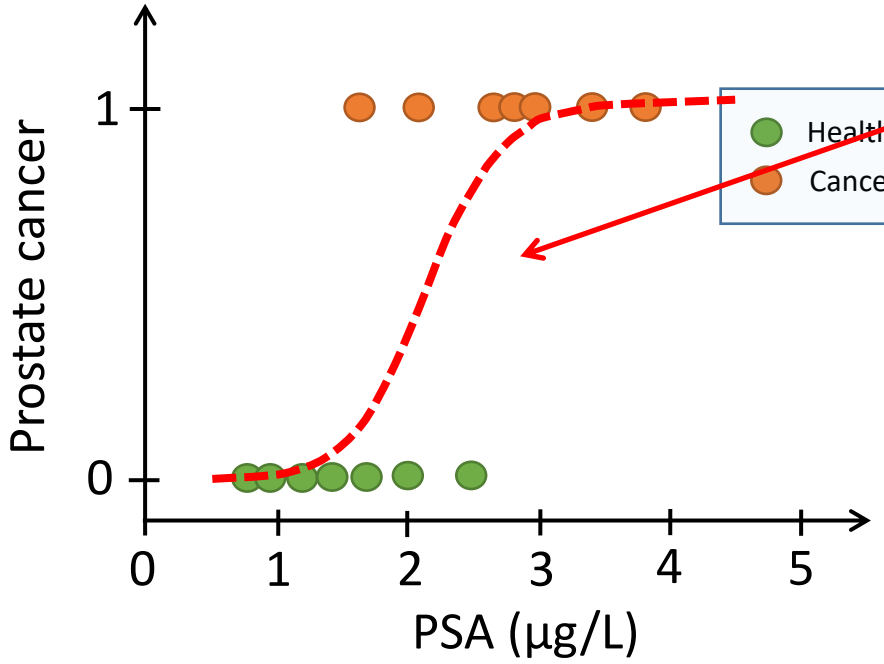


$$f(z) = \frac{1}{1 + e^{-(0.251)}} = 0.562$$

Logistic regression

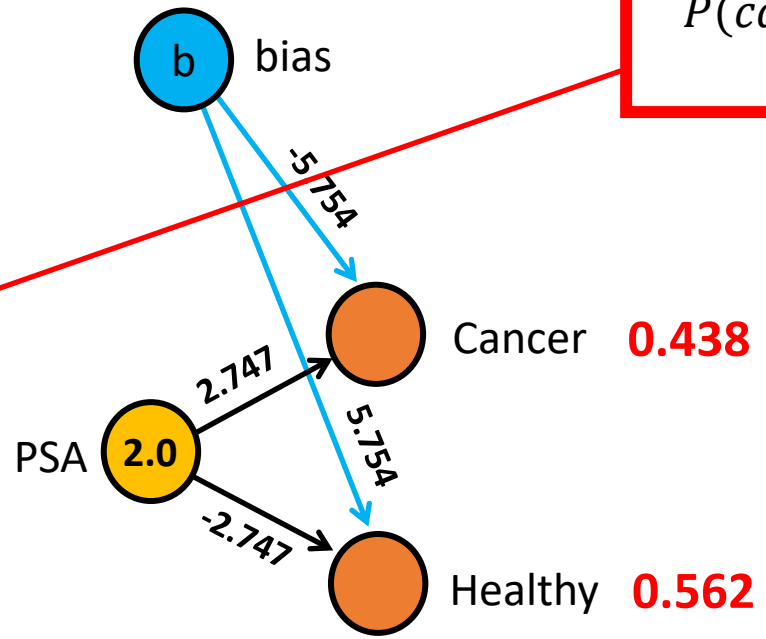
$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



Logistic regression

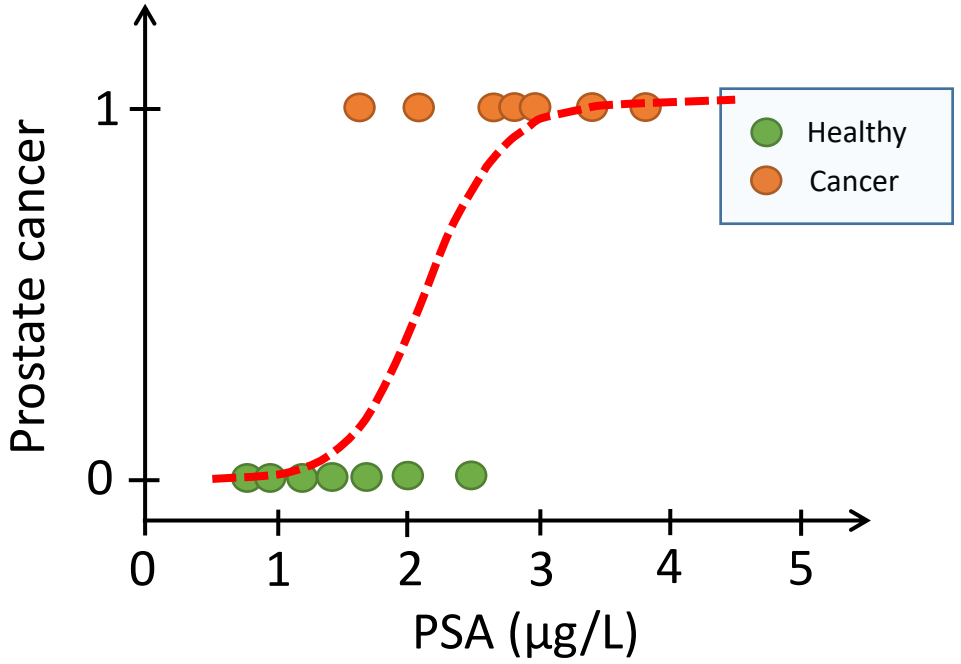
$$P(cancer) = \frac{1}{1 + e^{-(b_0 + b_1 PSA)}}$$

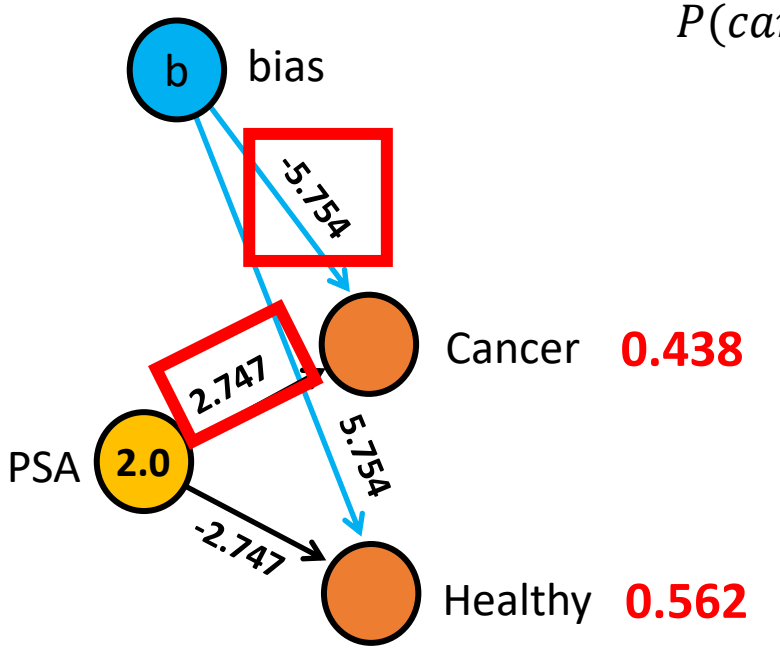


Logistic regression

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$





Logistic regression

$$P(cancer) = \frac{1}{1 + e^{-(b_0 + b_1 PSA)}}$$

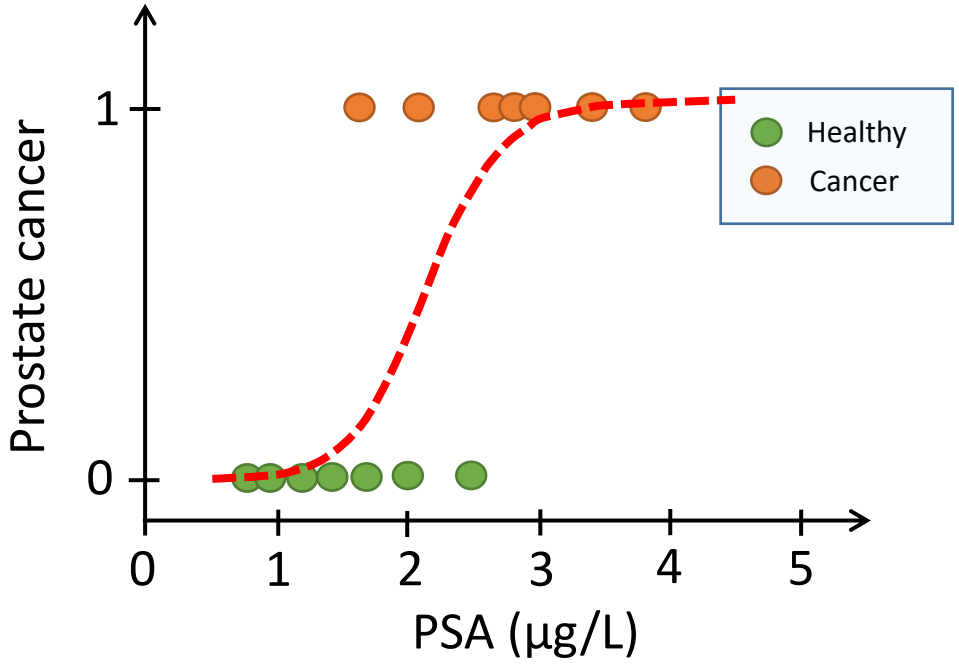
$$b_0 = -5.754$$

$$b_1 = 2.747$$

Logistic regression

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

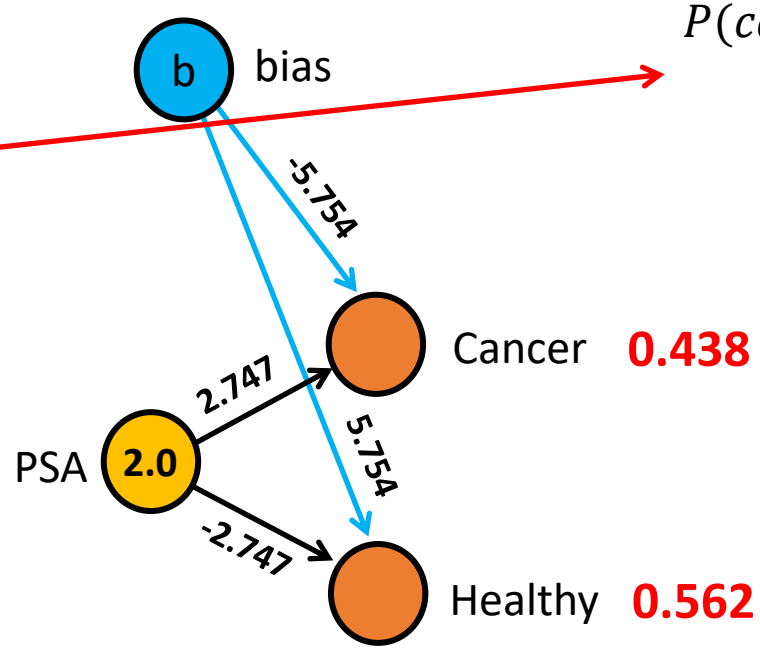


Logistic regression

$$P(cancer) = \frac{1}{1 + e^{-(b_0 + b_1 PSA)}}$$

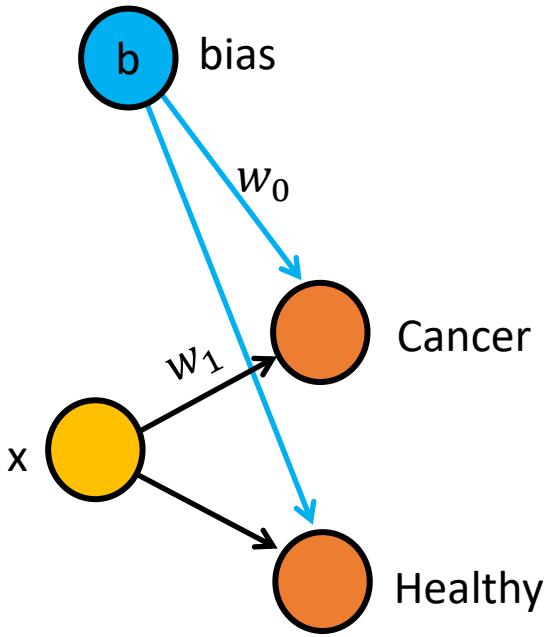
$b_0 = -5.754$

$b_1 = 2.747$



Different terminology

ANN



ANN	Regression
Input	Predictor or explanatory variable
Output	Response variable
Weights	Coefficients or parameters
bias	Intercept
Training	Fit a model or estimate the parameters
Backpropagation	Minimize the error
Loss function	Error function
Binary cross-entropy loss	Negative log-likelihood

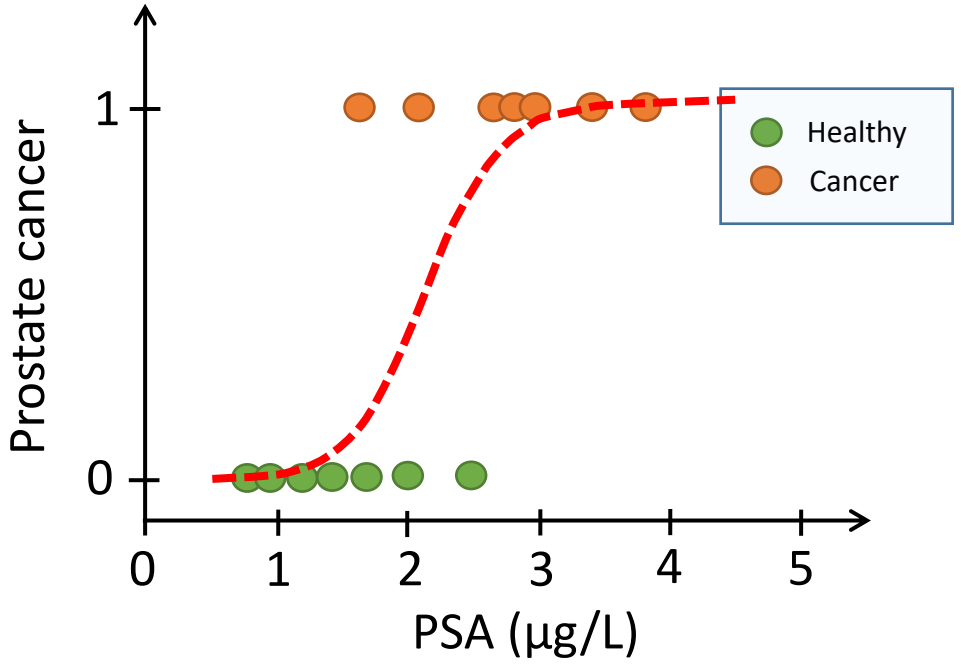
Logistic regression

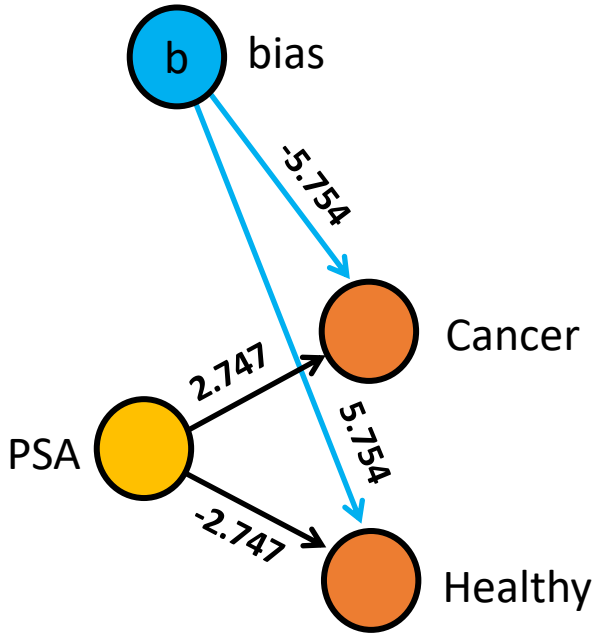
$$P(cancer) = \frac{1}{1 + e^{-(b_0 + b_1 x)}}$$

Predict training data

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



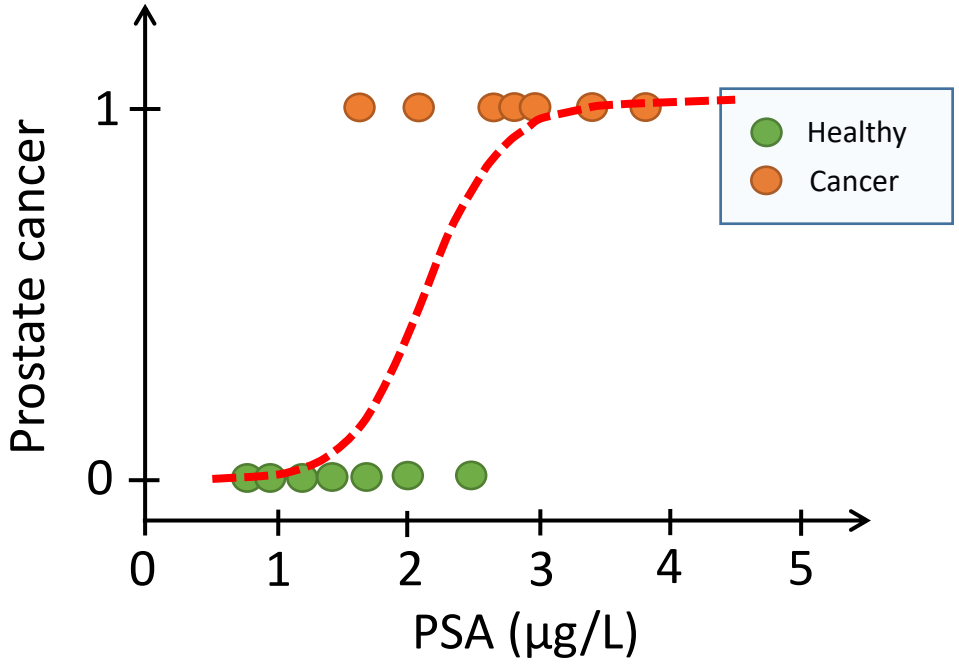


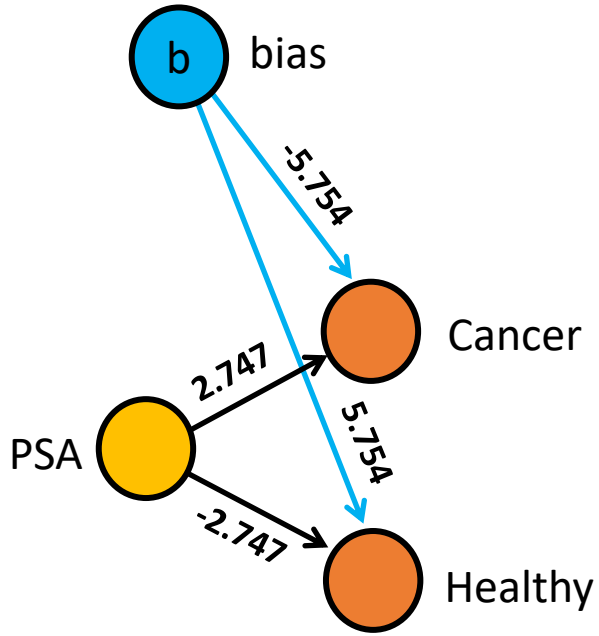
Status	PSA	Cancer	Prediction
Cancer	3.8	0.991	Cancer
Cancer	3.4	0.973	Cancer
Cancer	2.9	0.902	Cancer
Cancer	2.8	0.875	Cancer
Cancer	2.7	0.842	Cancer
Cancer	2.1	0.506	Cancer
Cancer	1.6	0.206	Healthy
Healthy	2.5	0.755	Cancer
Healthy	2.0	0.438	Healthy
Healthy	1.7	0.254	Healthy
Healthy	1.4	0.130	Healthy
Healthy	1.2	0.079	Healthy
Healthy	0.9	0.036	Healthy
Healthy	0.8	0.028	Healthy

Predict training data

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$





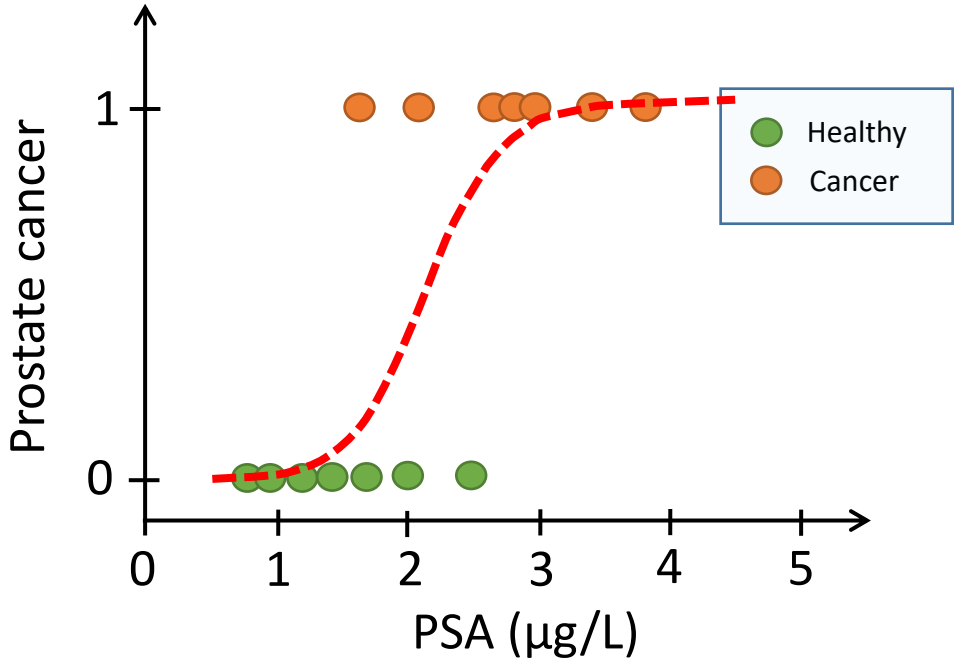
$$\text{Accuracy} = \frac{12}{14} = 86\%$$

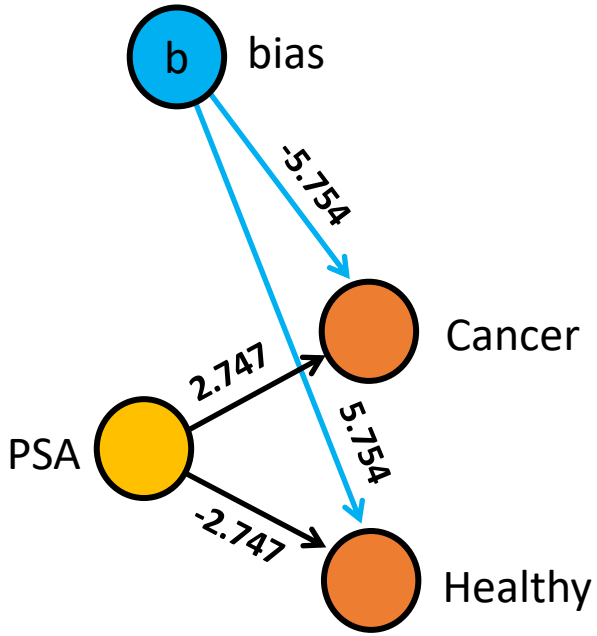
Status	PSA	Cancer	Prediction
Cancer	3.8	0.991	Cancer
Cancer	3.4	0.973	Cancer
Cancer	2.9	0.902	Cancer
Cancer	2.8	0.875	Cancer
Cancer	2.7	0.842	Cancer
Cancer	2.1	0.506	Cancer
Cancer	1.6	0.206	Healthy
Healthy	2.5	0.755	Cancer
Healthy	2.0	0.438	Healthy
Healthy	1.7	0.254	Healthy
Healthy	1.4	0.130	Healthy
Healthy	1.2	0.079	Healthy
Healthy	0.9	0.036	Healthy
Healthy	0.8	0.028	Healthy

Validation

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



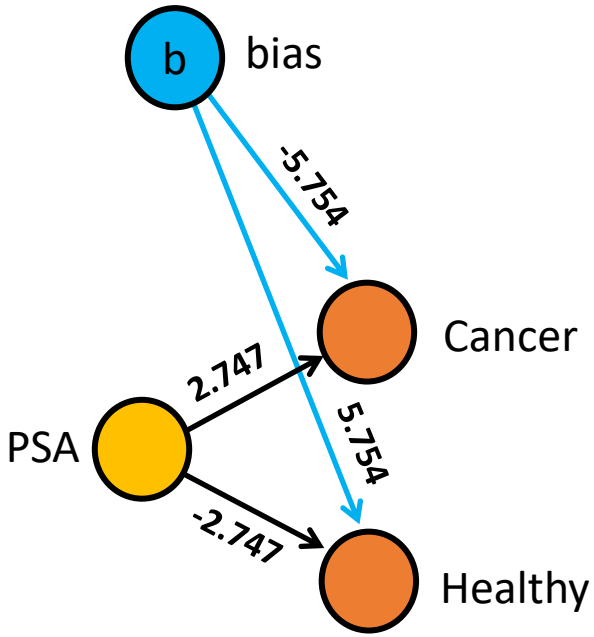


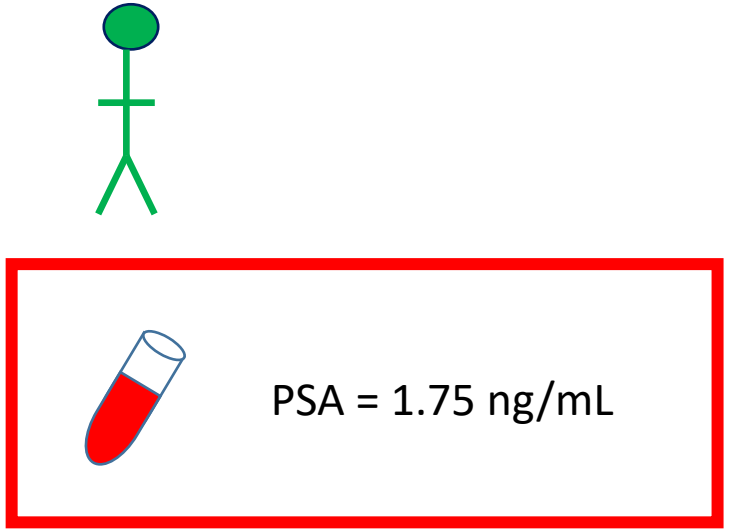
Test data set

or

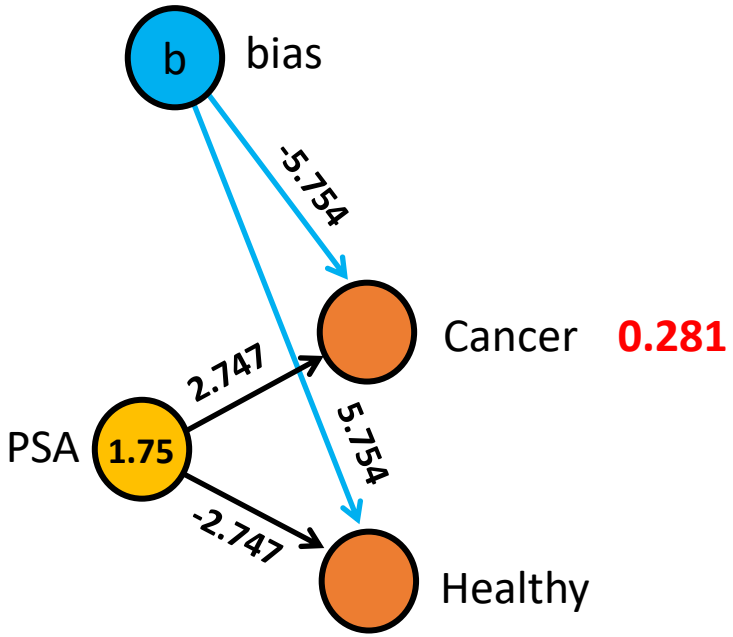
Cross-validation

Prediction





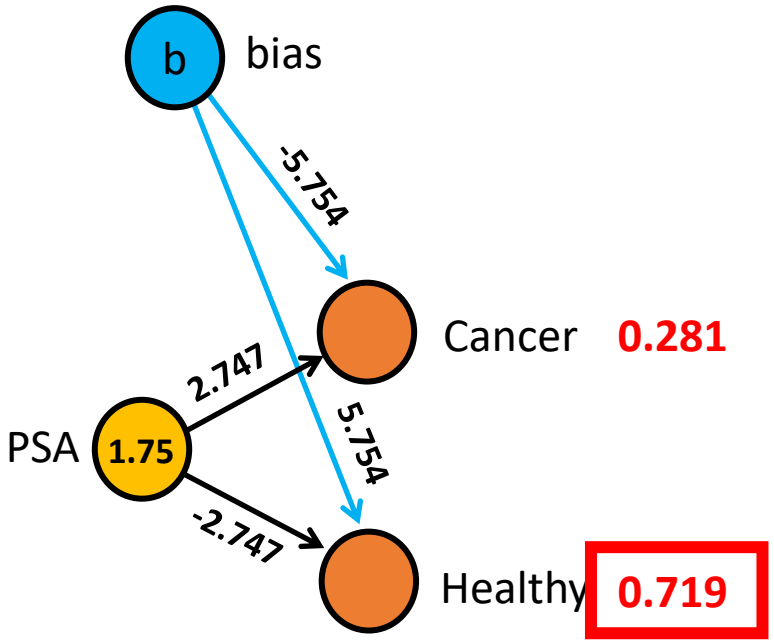
Prediction



$$z = -5.754 + 2.747 \cdot 1.75 = -0.93775$$

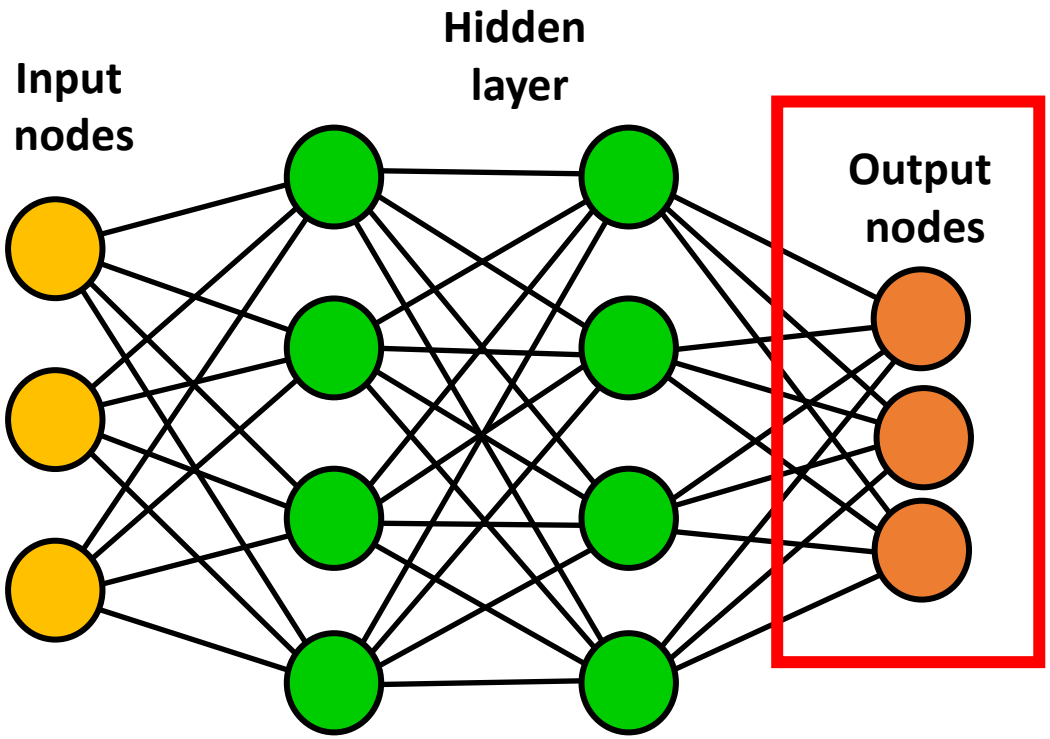
$$f(z) = \frac{1}{1 + e^{-(-0.93775)}} = \boxed{0.281}$$

Prediction



$$1 - 0.281 = 0.719$$

More than two output nodes

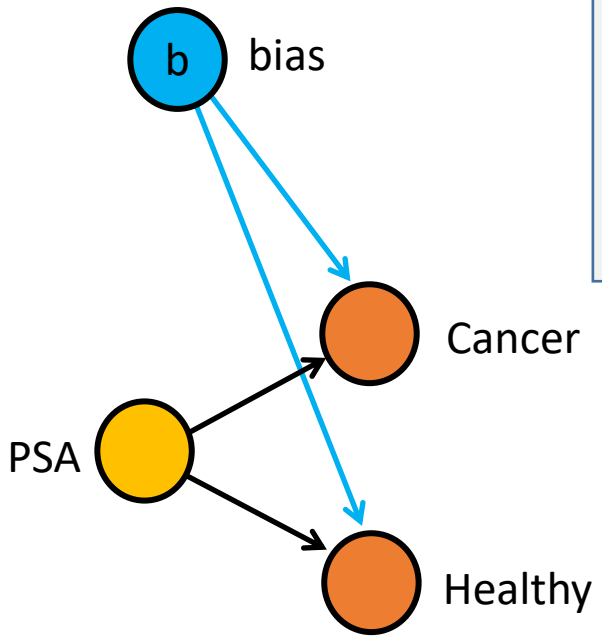


Softmax

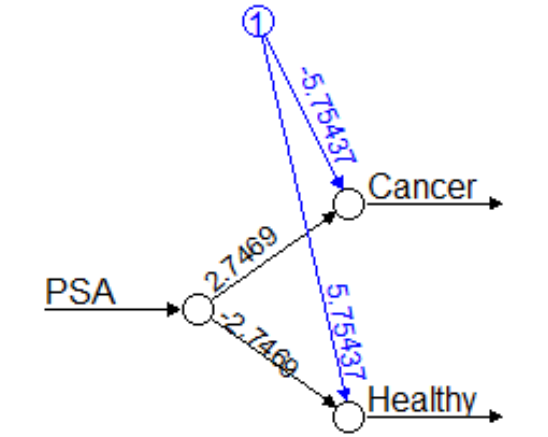
$$s(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

R code

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8




```
PSA=c(3.8, 3.4, 2.9, 2.8, 2.7, 2.1, 1.6,2.5, 2.0, 1.7, 1.4, 1.2, 0.9, 0.8)
Group=factor(rep(c("Cancer","Healthy"),c(7,7)),levels=c("Healthy","Cancer"))
df=data.frame(Group,PSA)
install.packages("neuralnet")
library(neuralnet)
nn=neuralnet(Group~PSA,data=df, hidden=0, act.fct = "logistic",
          linear.output = FALSE, err.fct="ce", threshold = 1e-6)
print(nn)
plot(nn)
df2=data.frame(PSA=2)
predict(nn,df2)
```

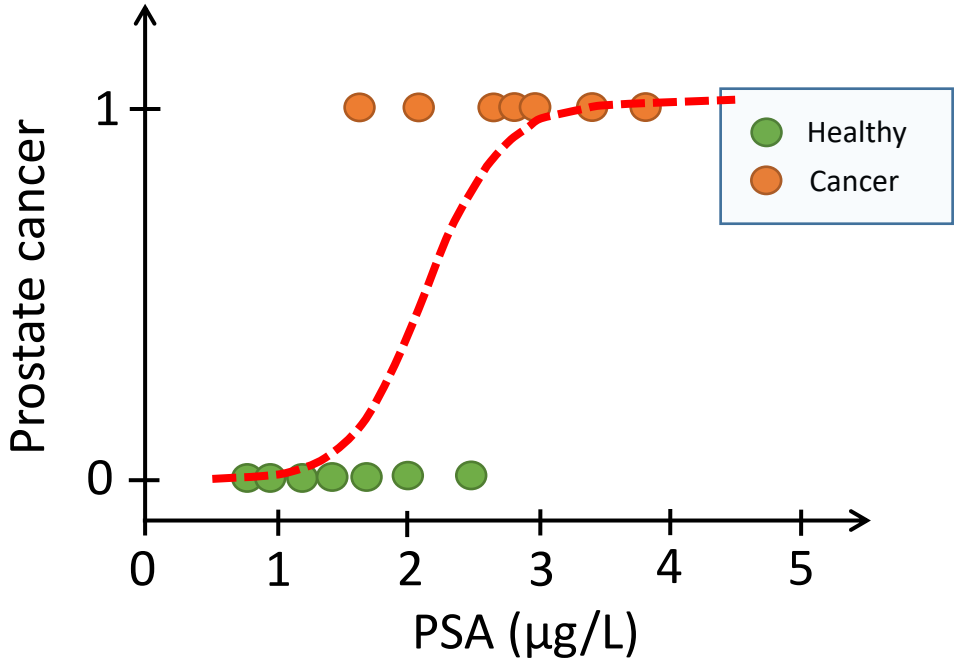


Error: 10.535597 Steps: 754

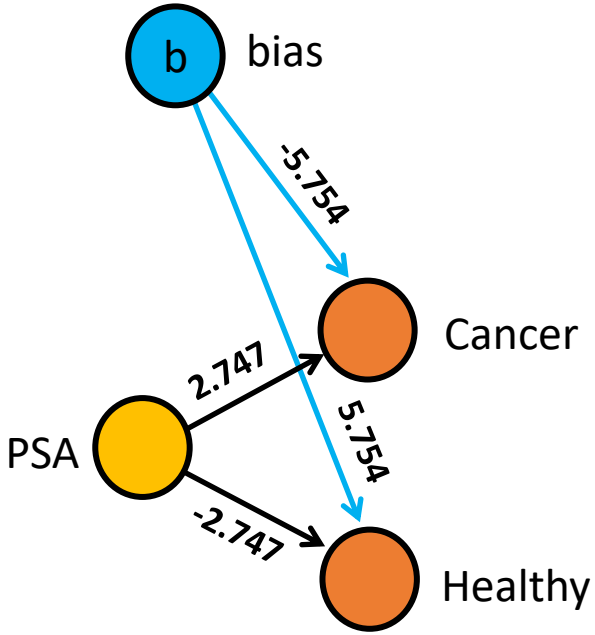
Estimating the weights

Where do the values for the weights come from?

Estimating the weights

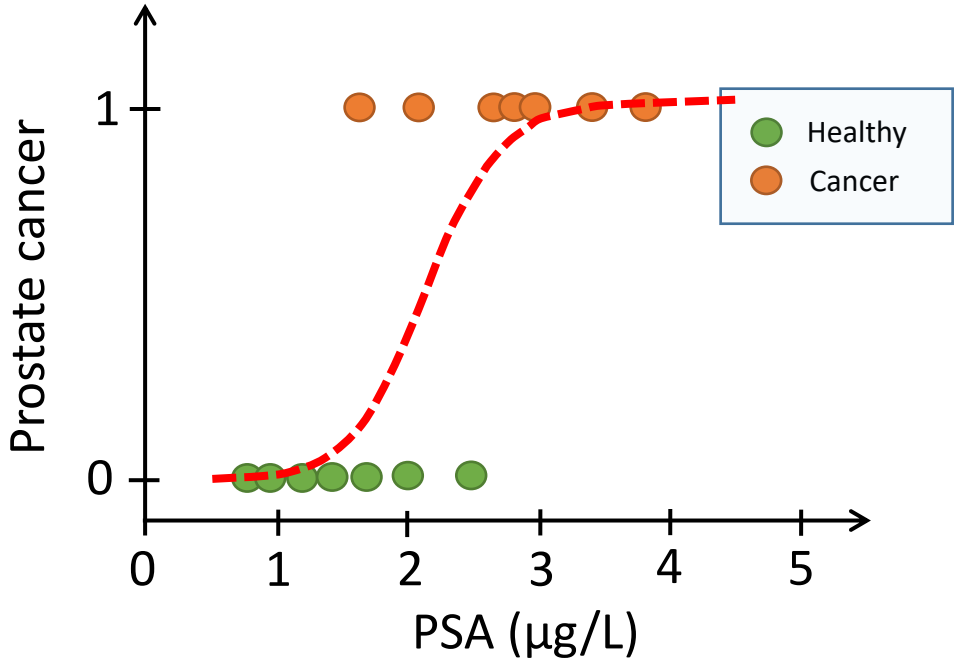


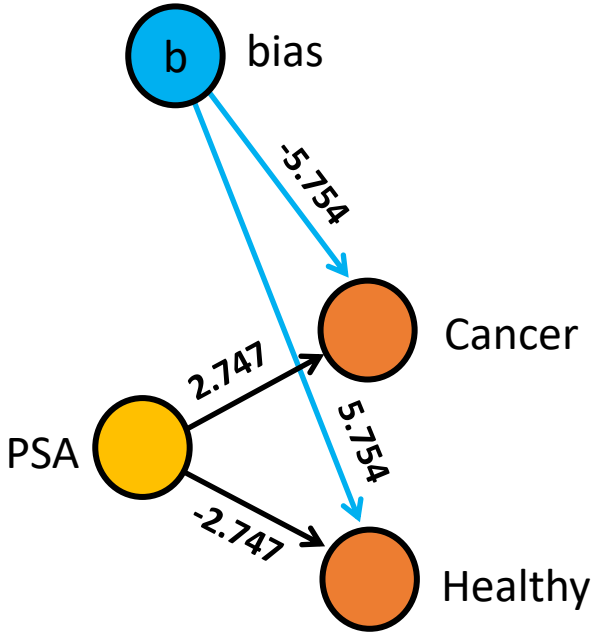
MLE
Least squares



Estimating the weights

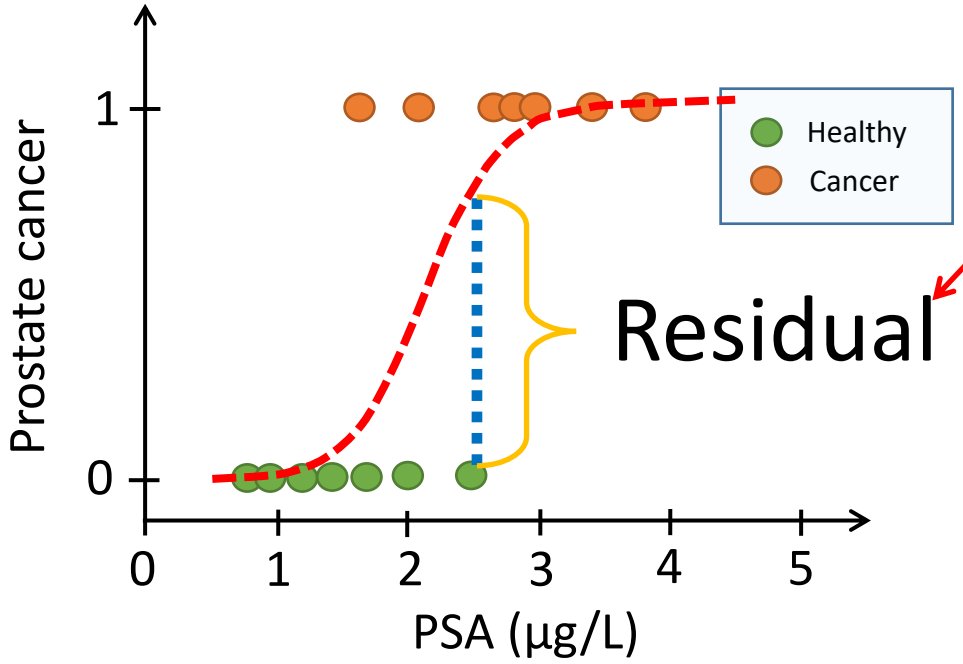
$$\mathrm{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2$$

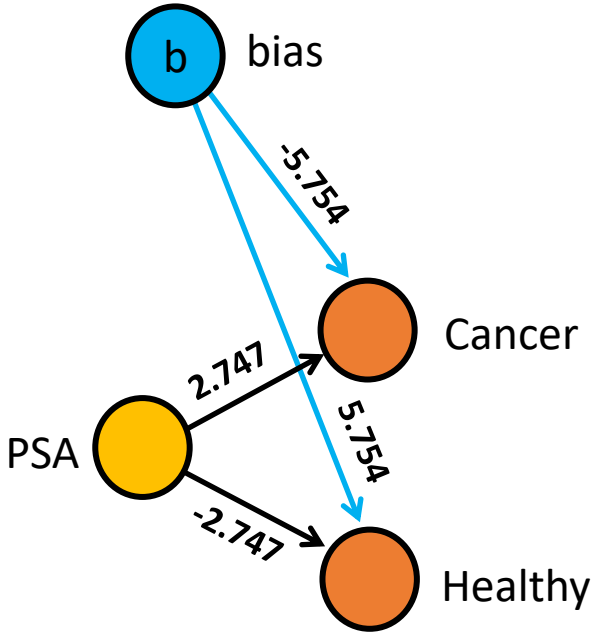




Estimating the weights

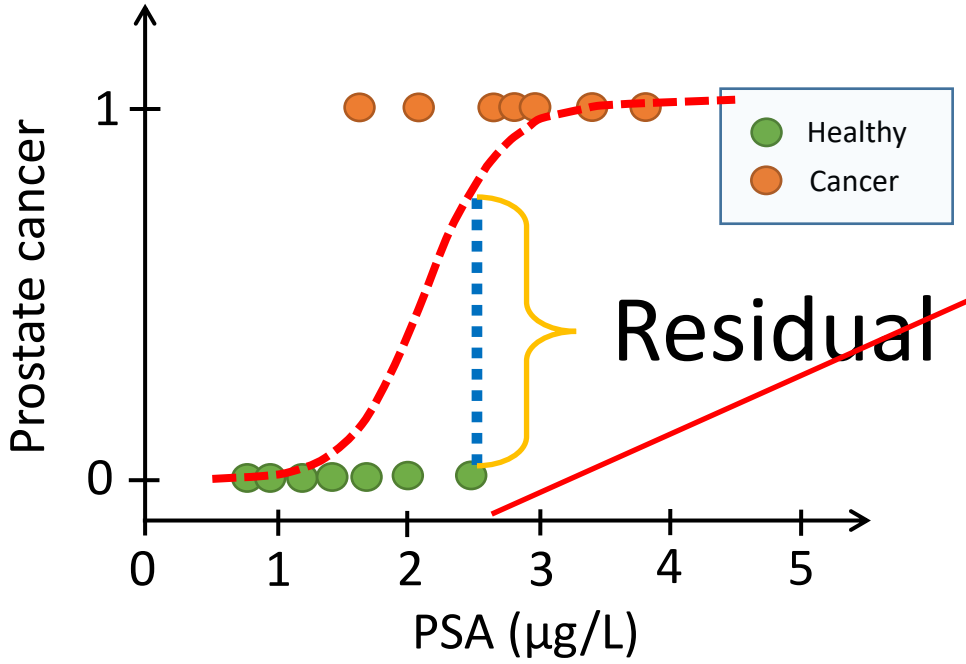
$$\mathrm{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2$$





Estimating the weights

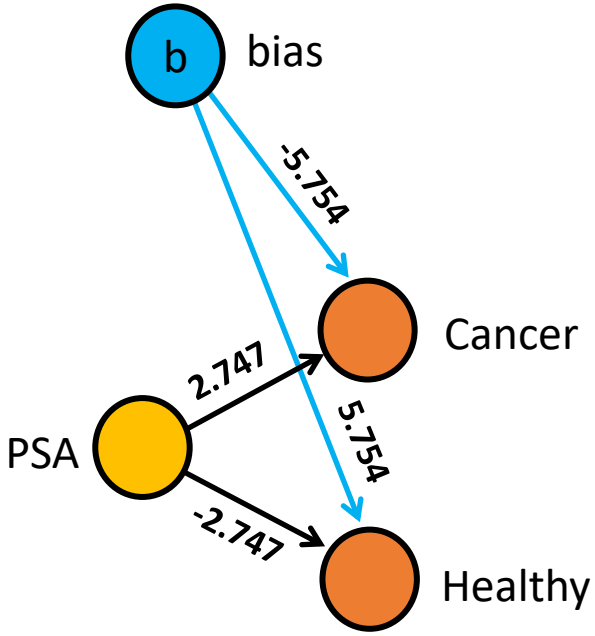
$$\mathrm{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2$$



$$z = -5.754 + 2.747 \cdot 2.5 = 1.12$$

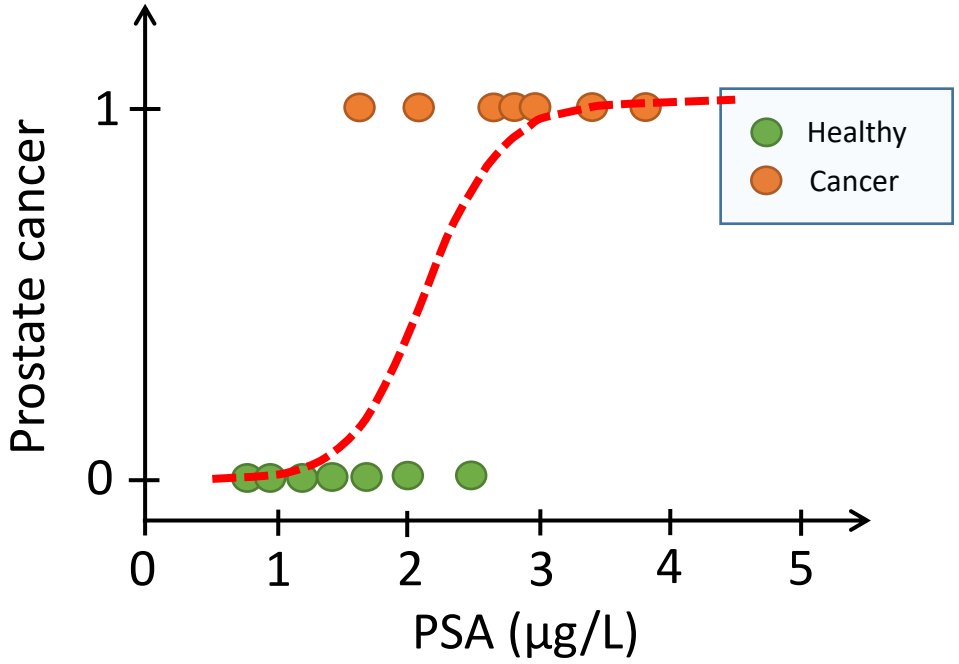
$$f(z) = \frac{1}{1 + e^{-1.12}} = 0.75$$

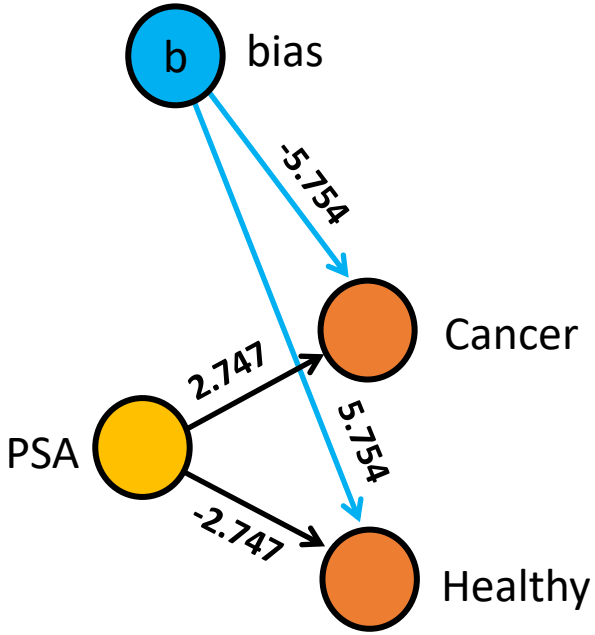
$$y_i - \hat{y}_i = 0 - 0.75 = -0.75$$



Estimating the weights

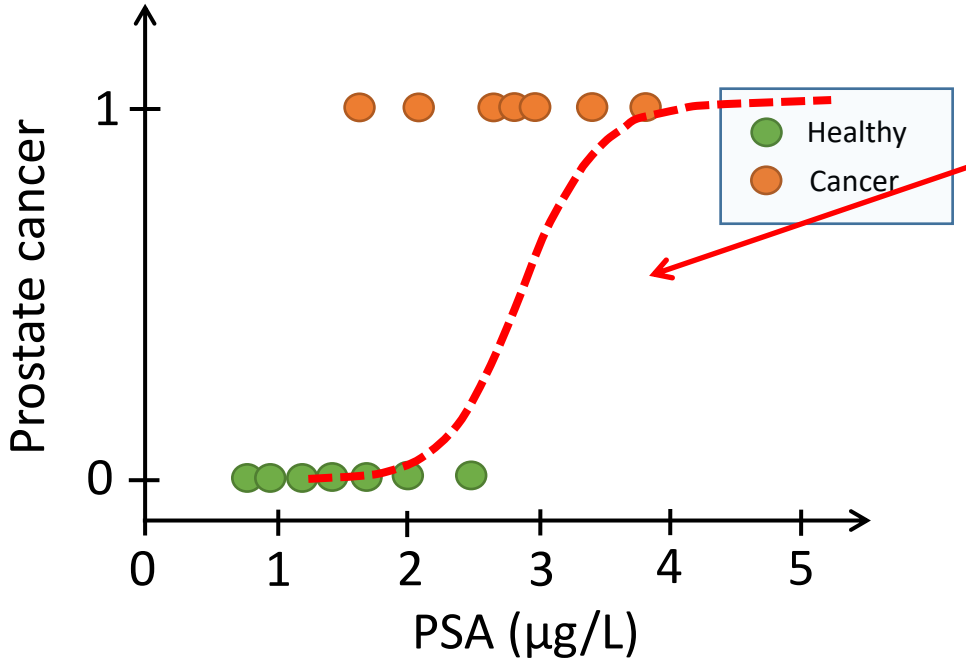
$$\text{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2 = 1.78$$

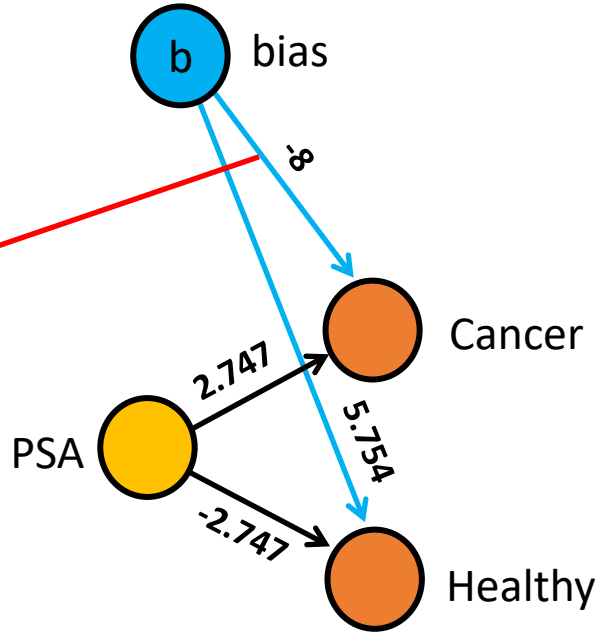




Estimating the weights

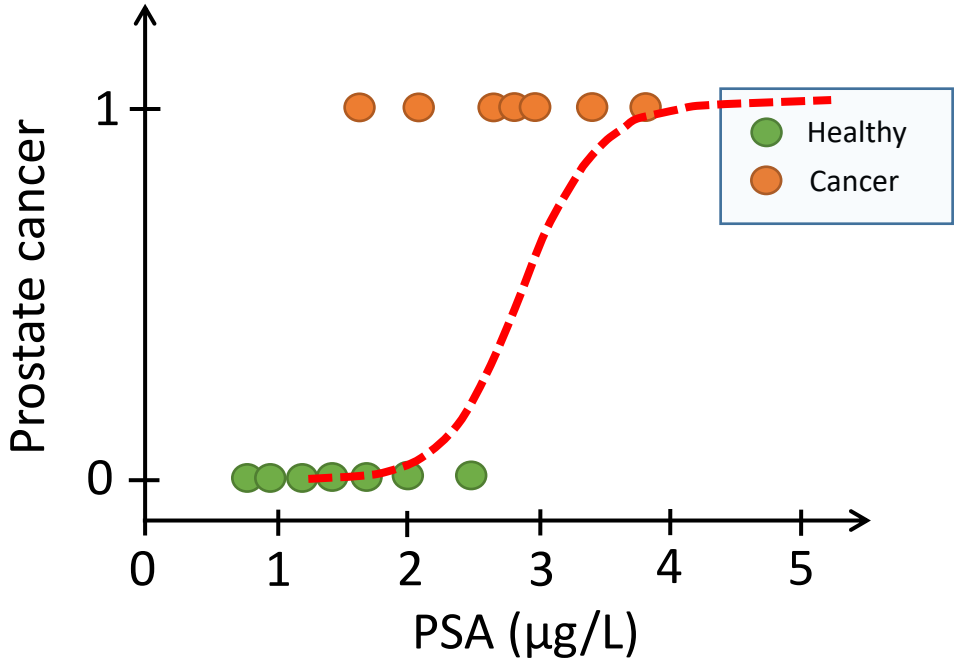
$$\mathrm{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2 =$$

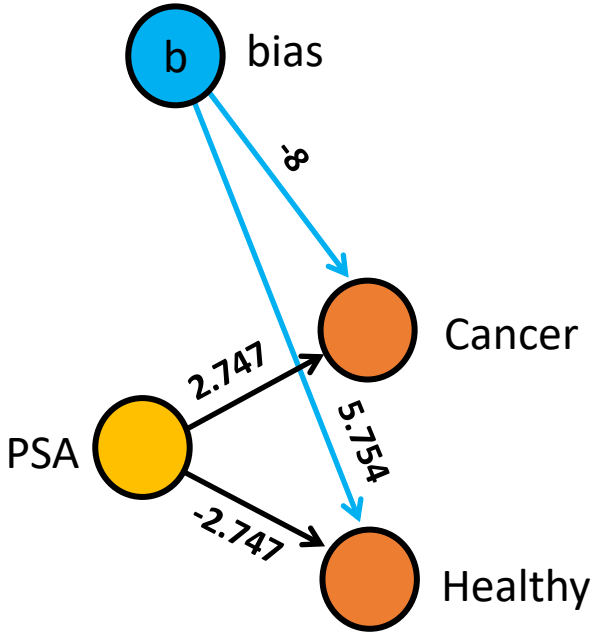




Estimating the weights

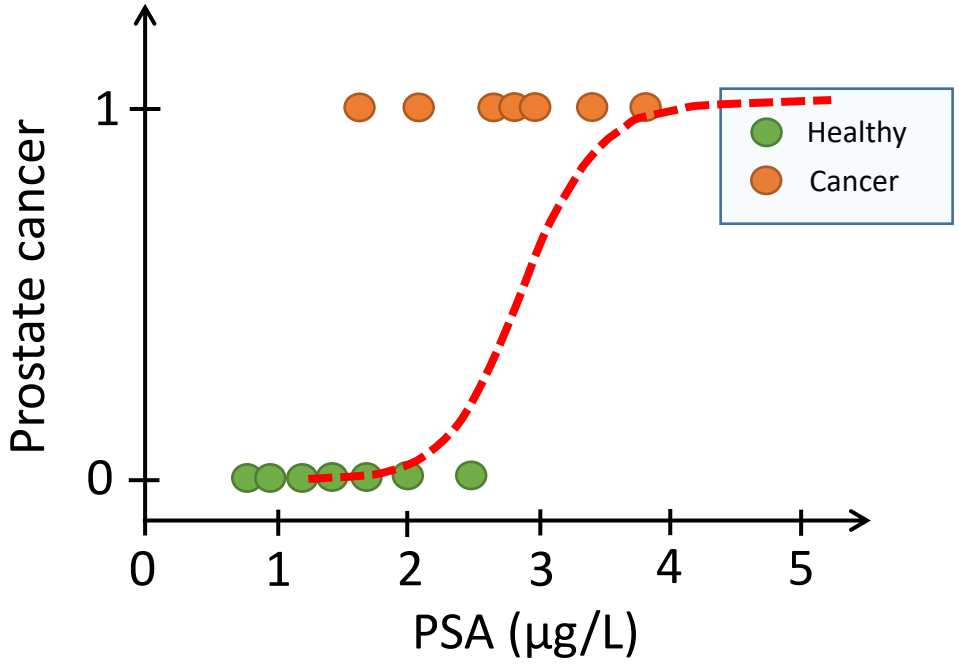
$$\text{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2 = 2.88$$

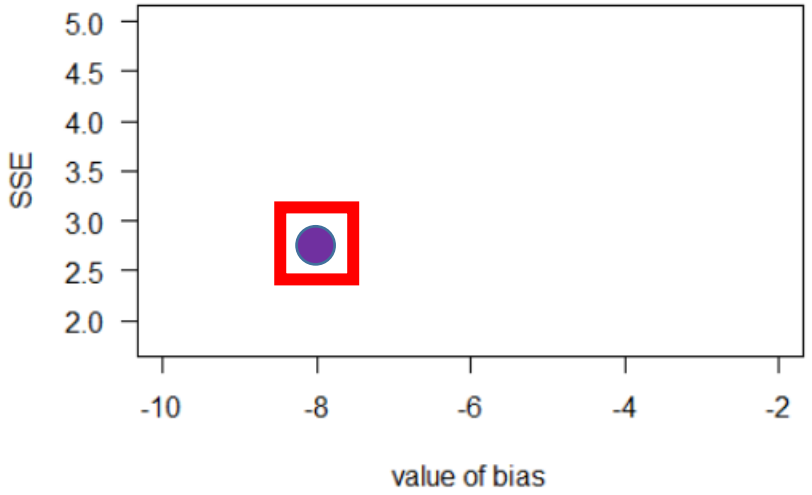


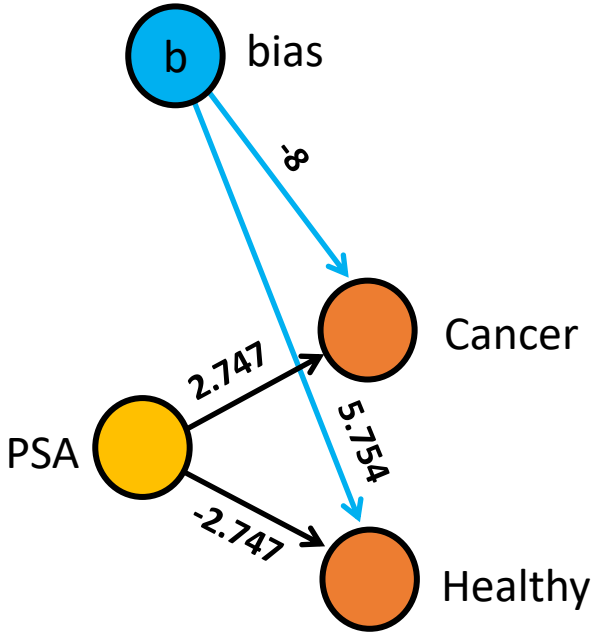


Estimating the weights

$$\text{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2 = 2.88$$

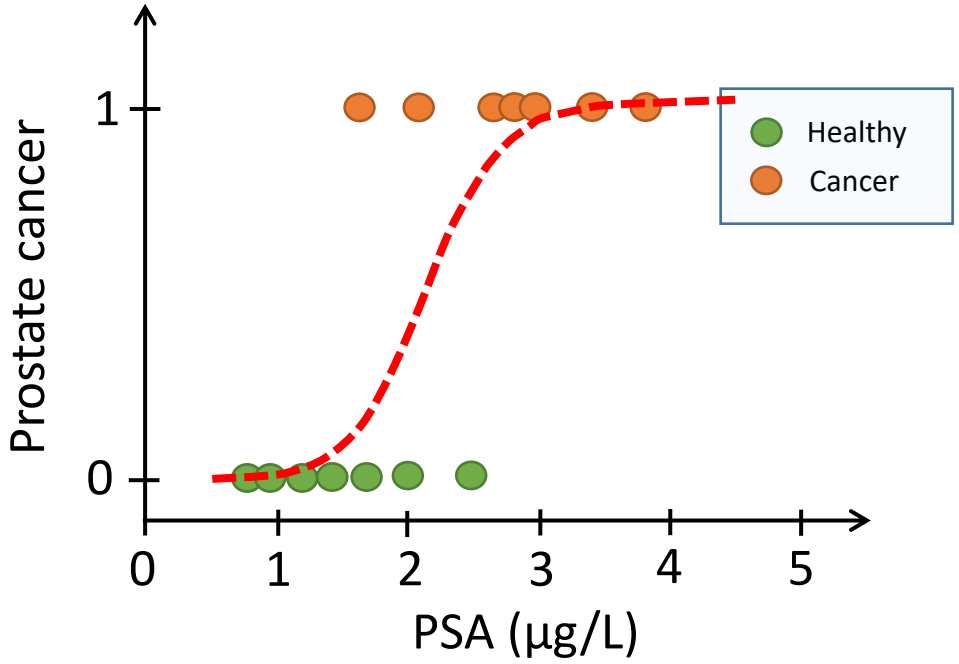


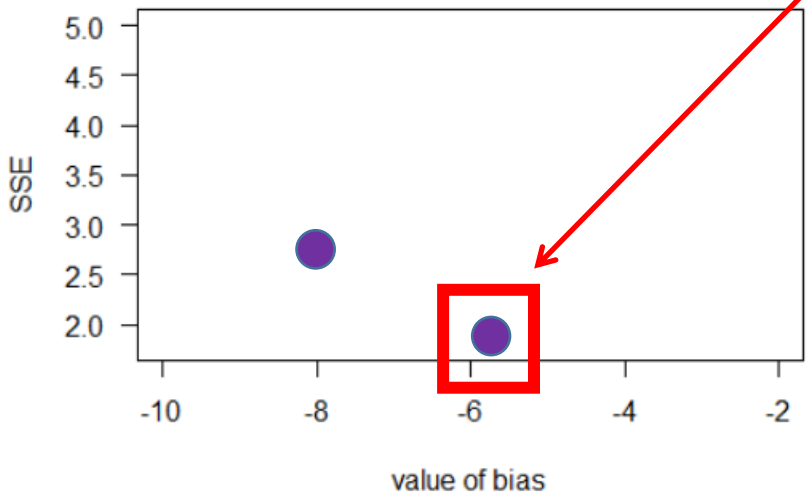


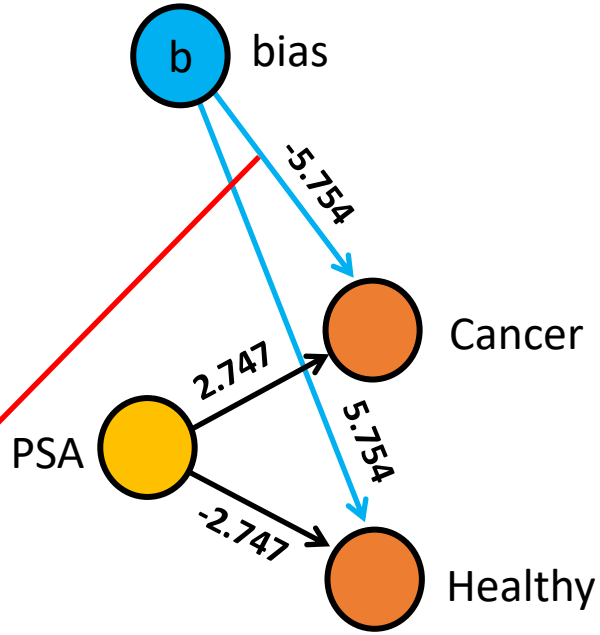


Estimating the weights

$$\text{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2 = 1.78$$

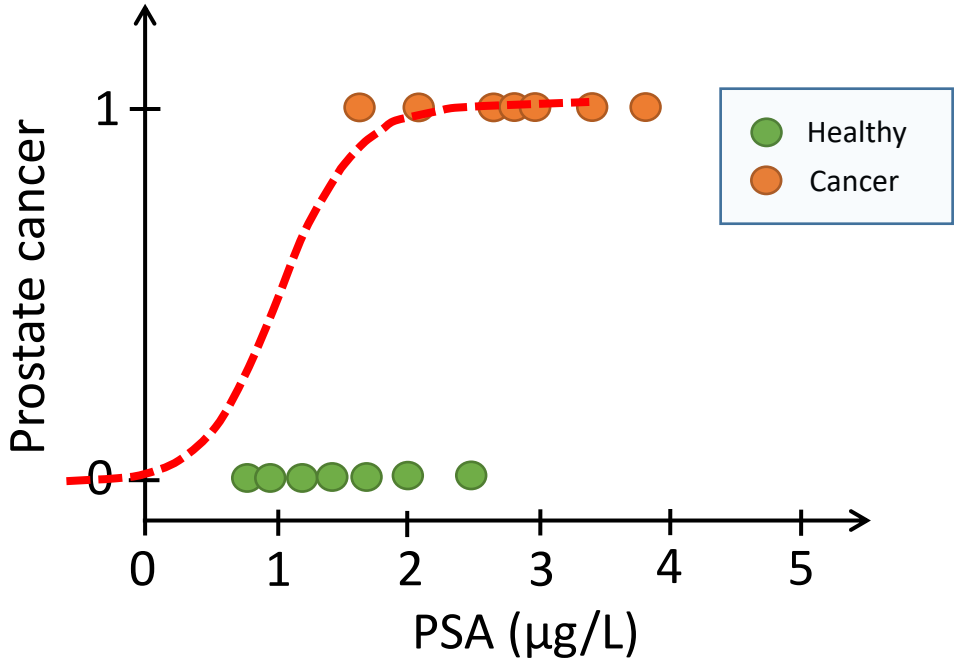


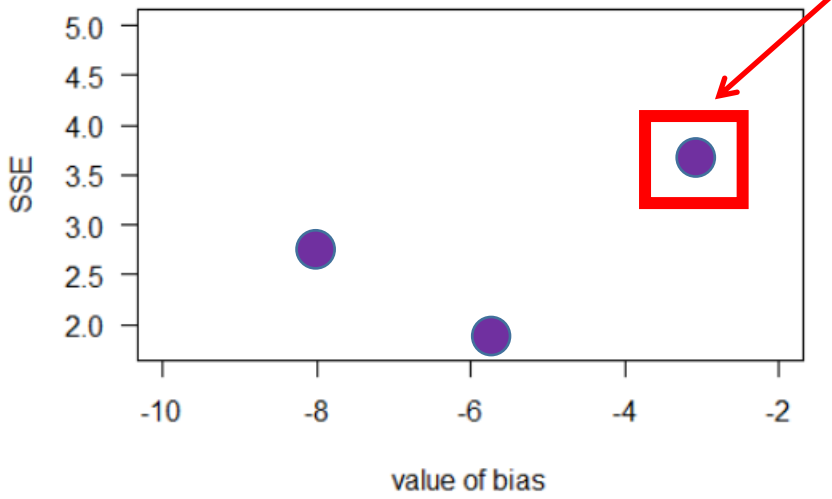


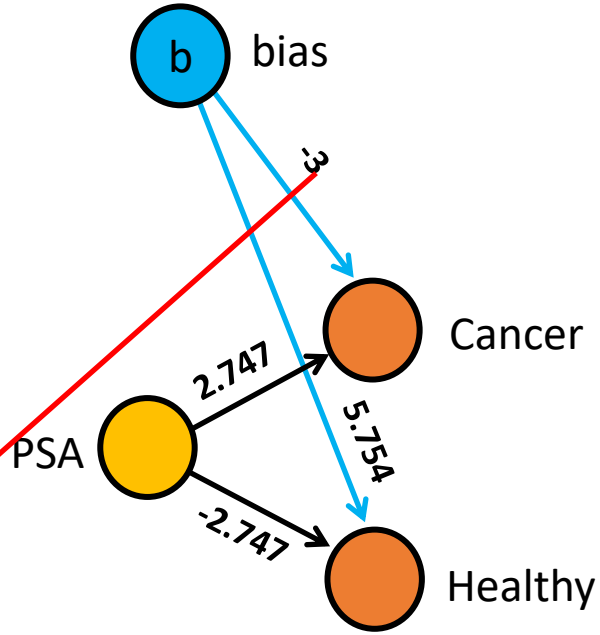


Estimating the weights

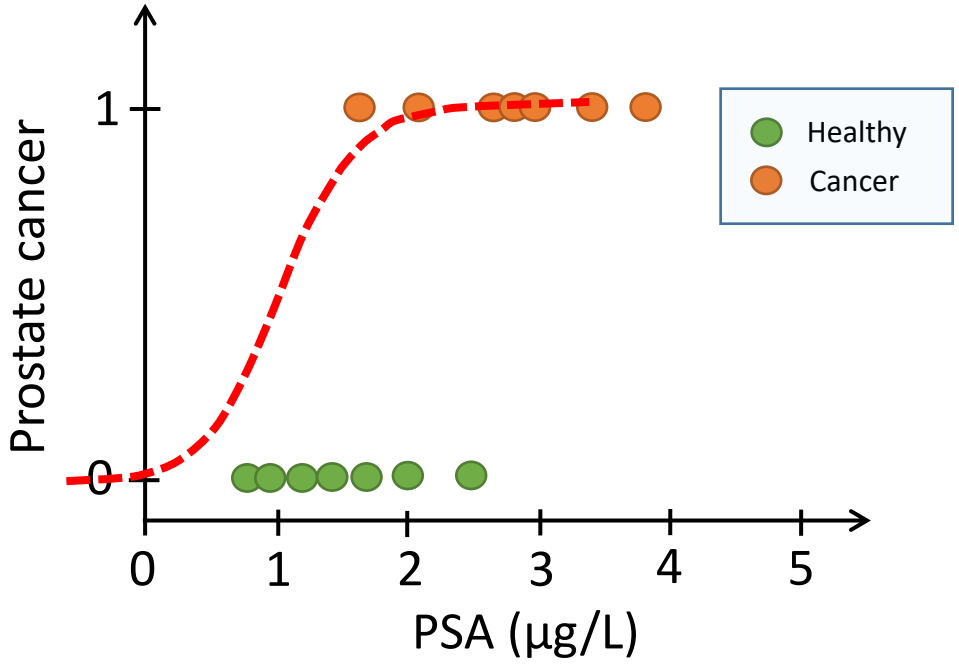
$$\text{SSE} = \sum_{i}^{n} (y_i - \hat{y}_i)^2 = 3.62$$

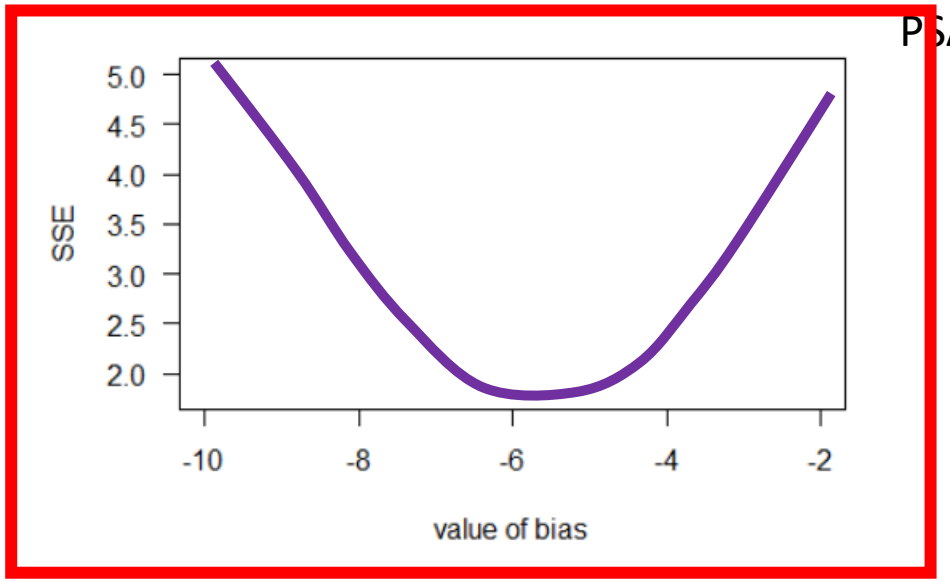


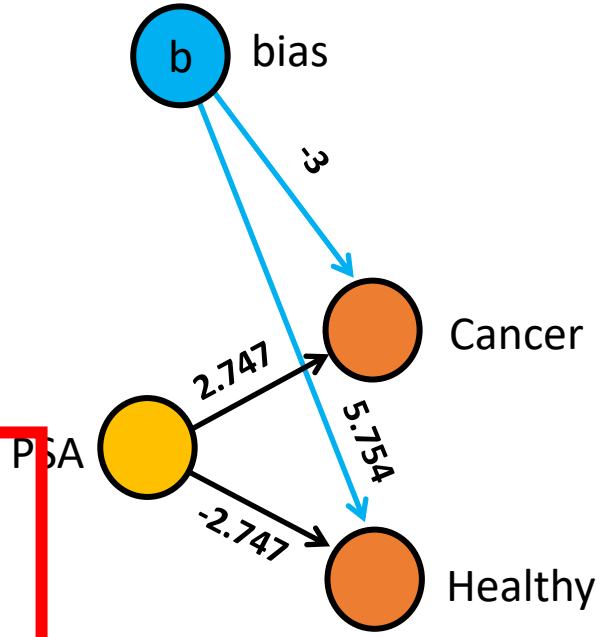




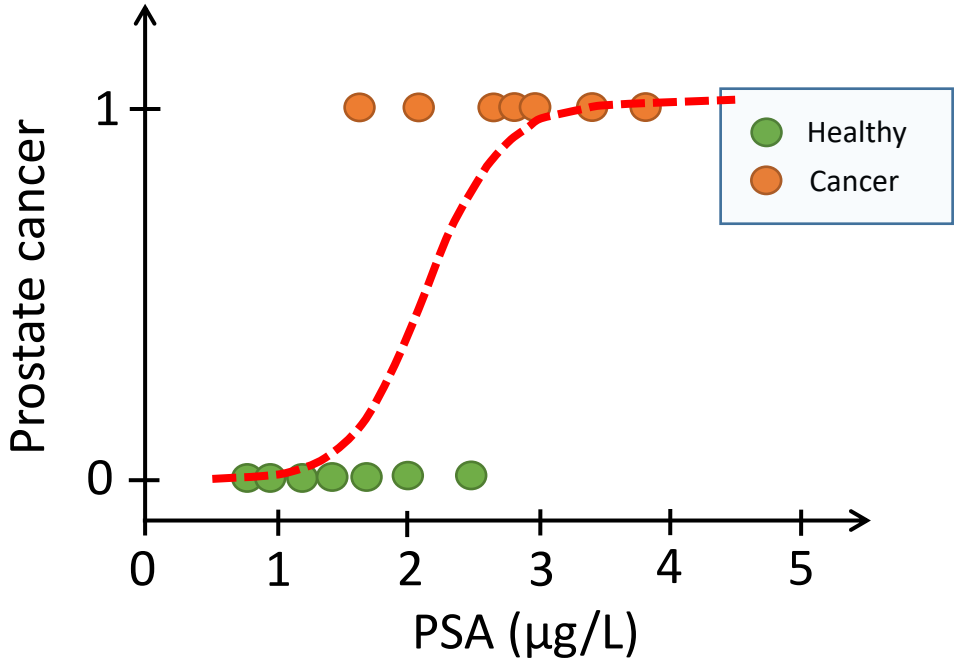
Estimating the weights

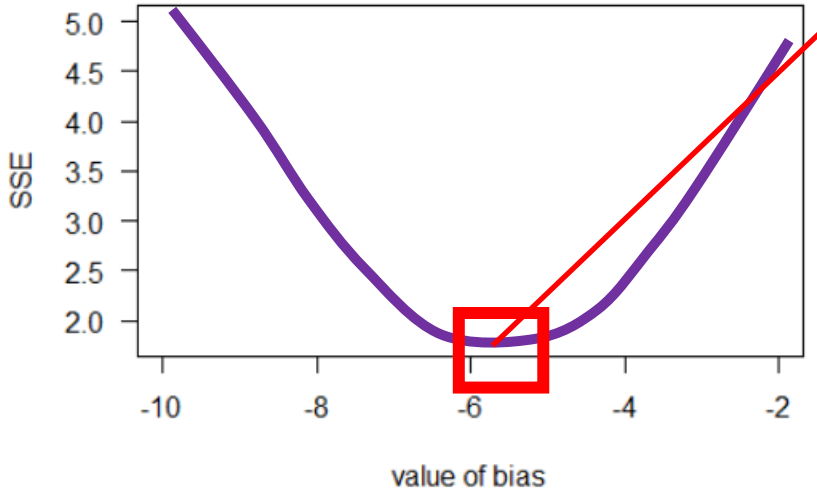


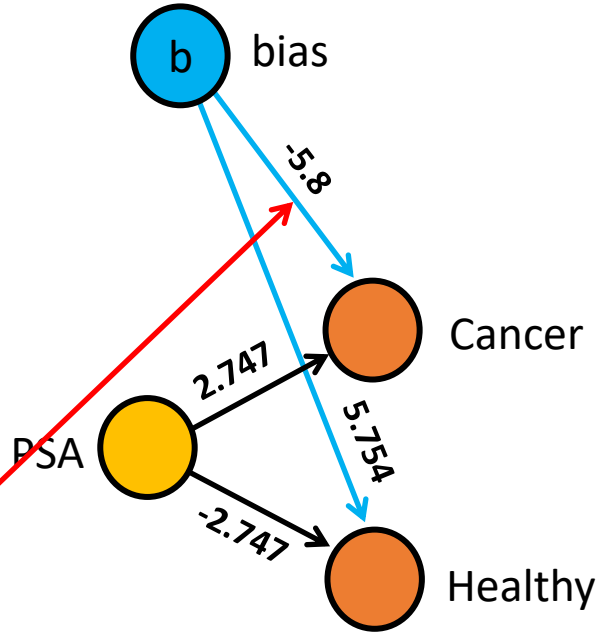




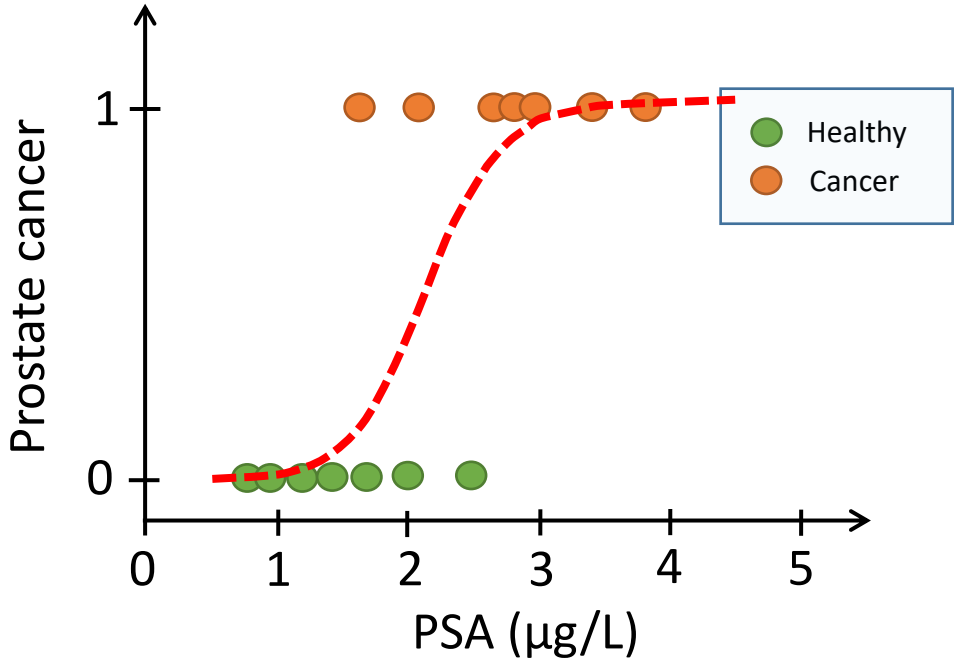
Estimating the weights



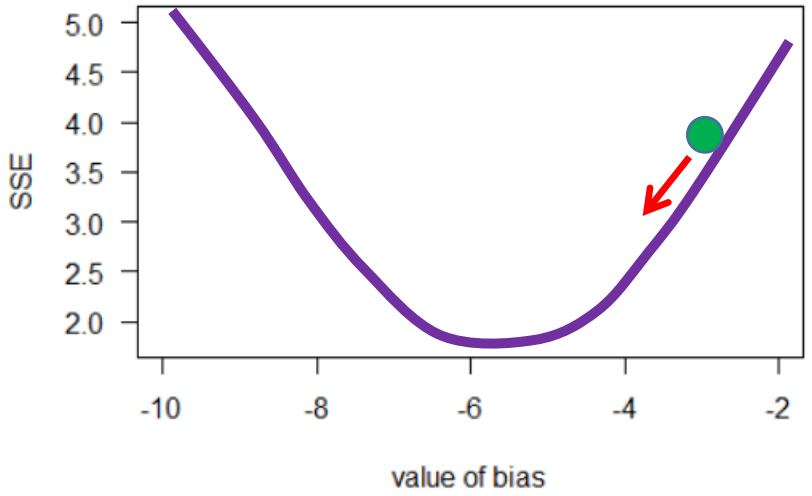


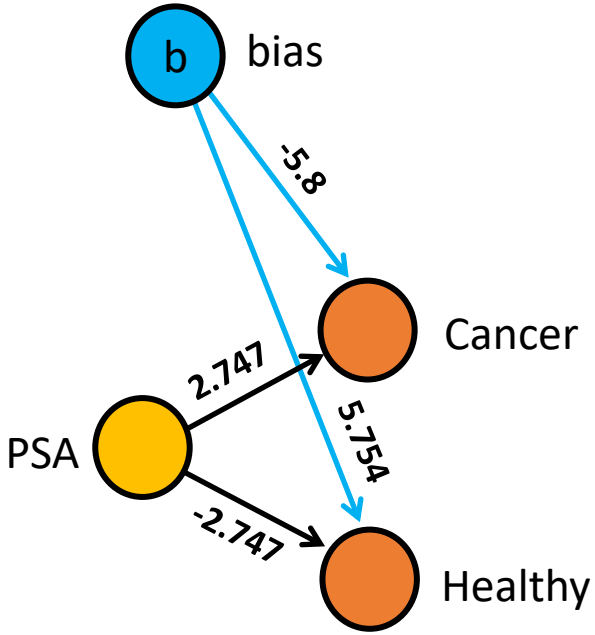


Guess initial values of the weights

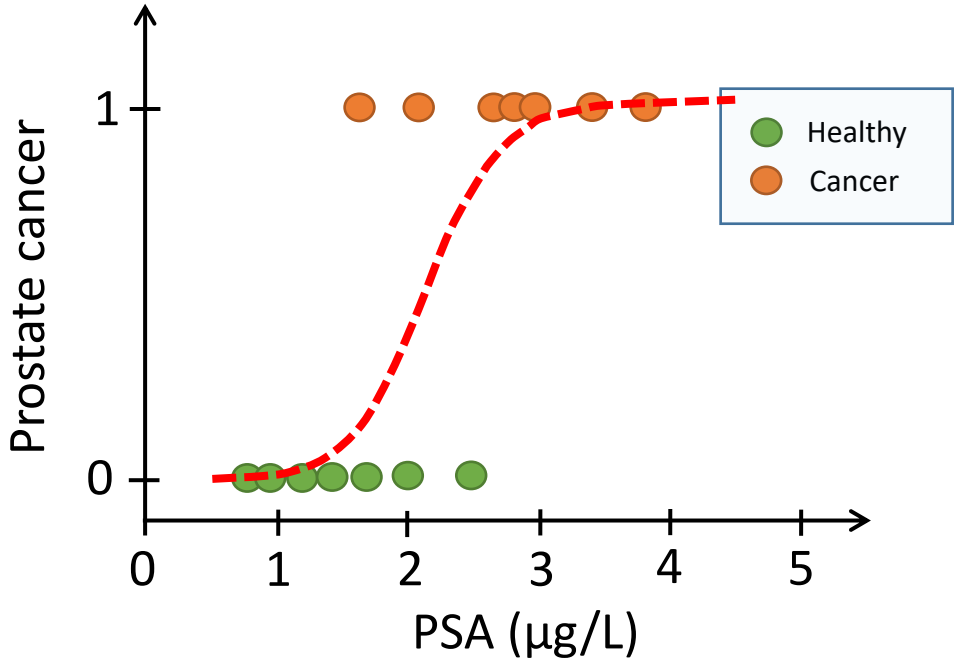


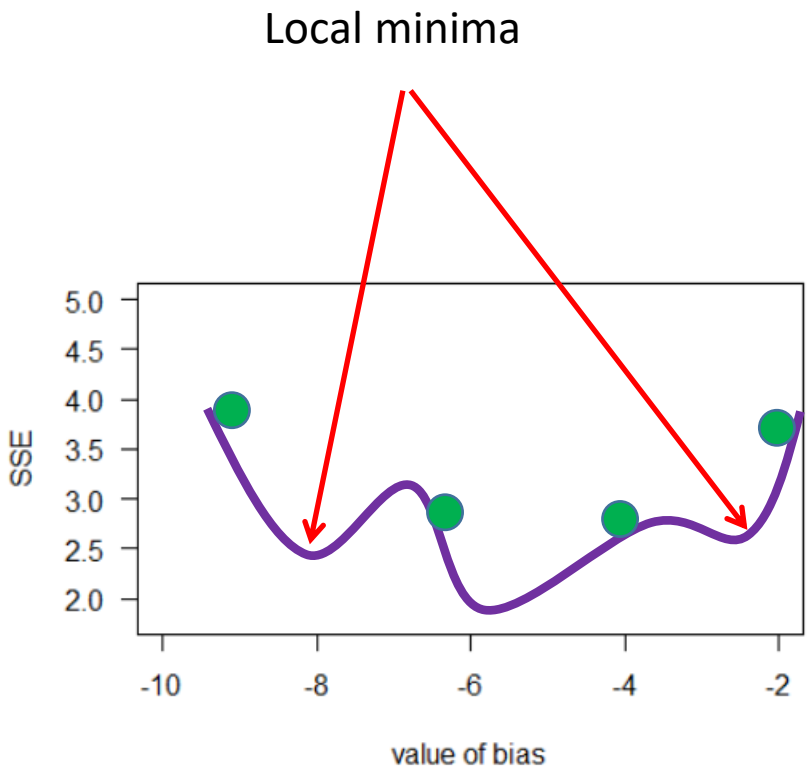
Steepest descent

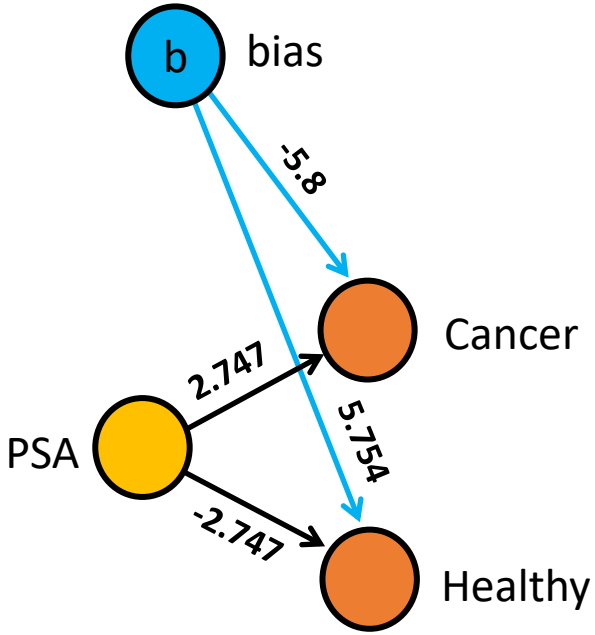




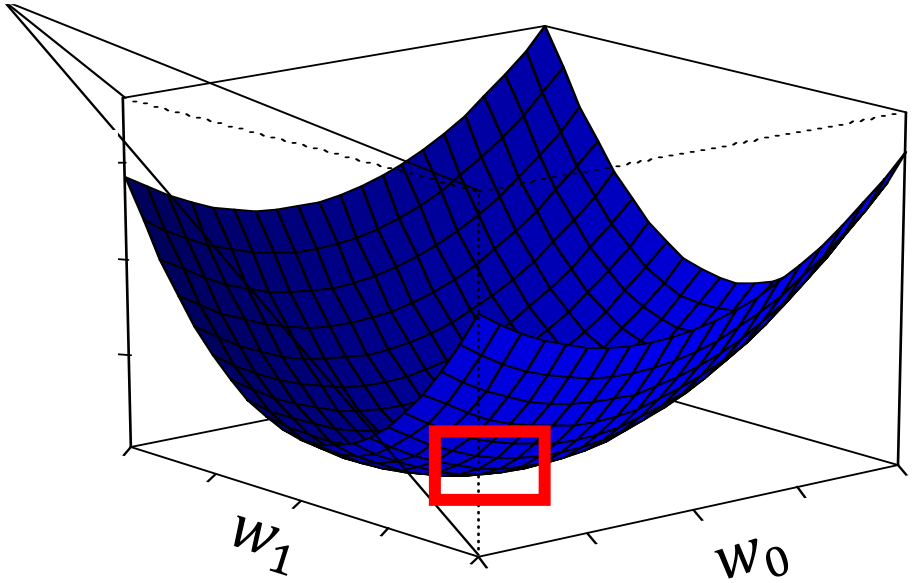
Guess initial values of the weights

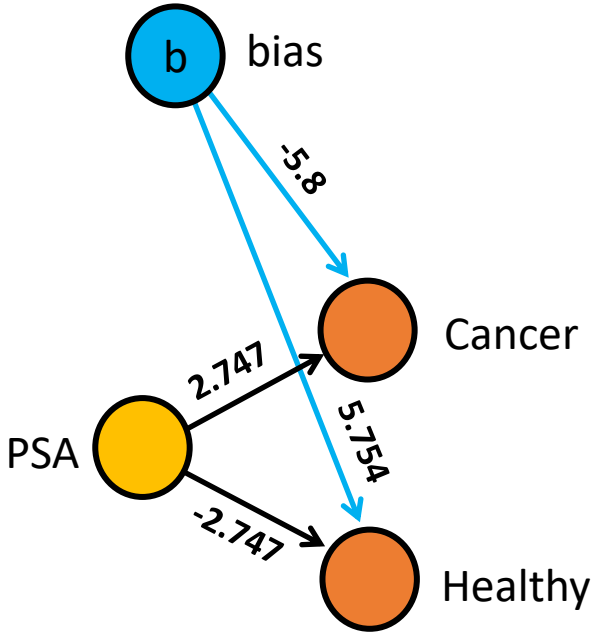






Estimating the weights

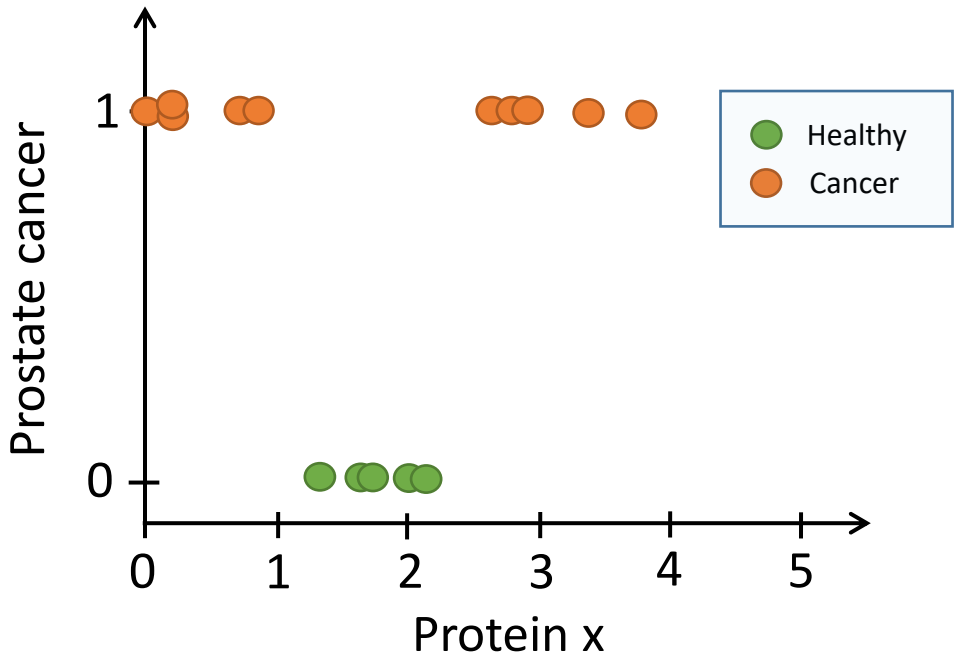




The hidden layer

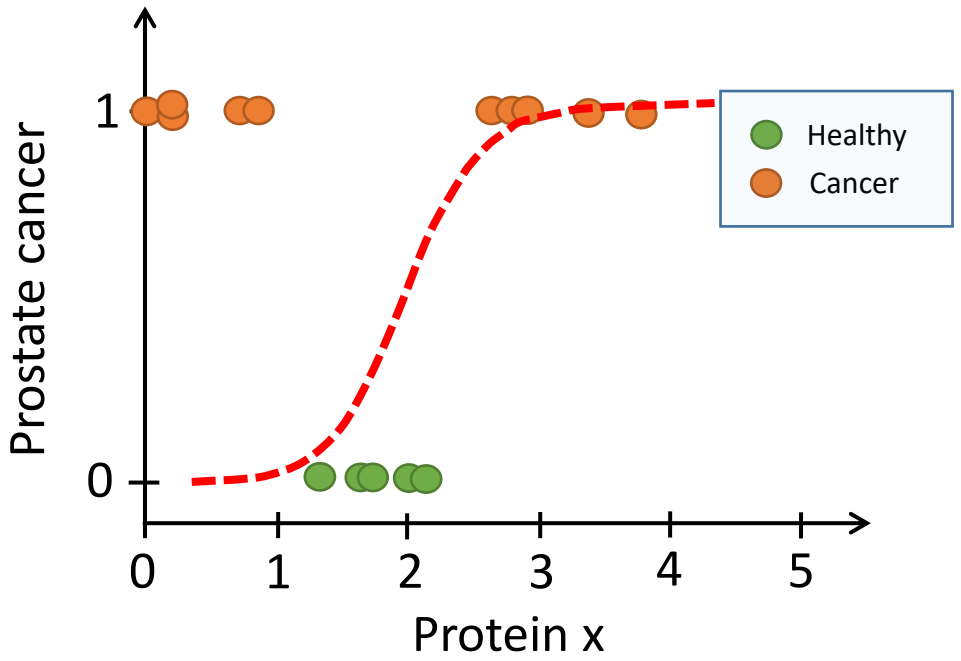
The hidden layer

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1



The hidden layer

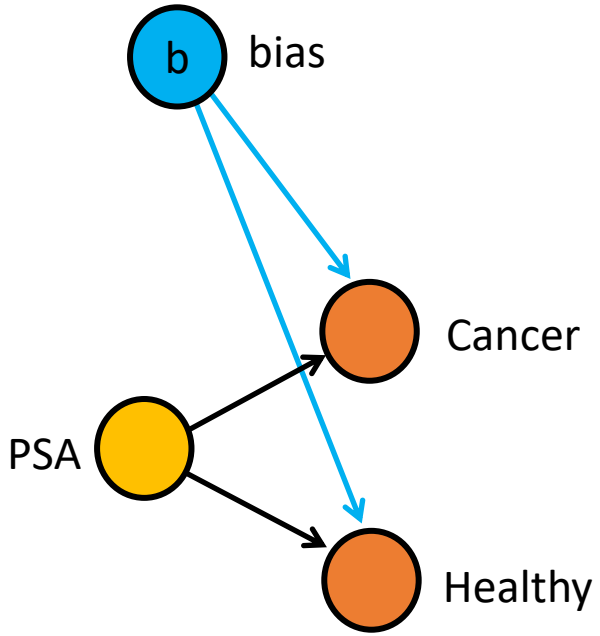
Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1



Logistic regression

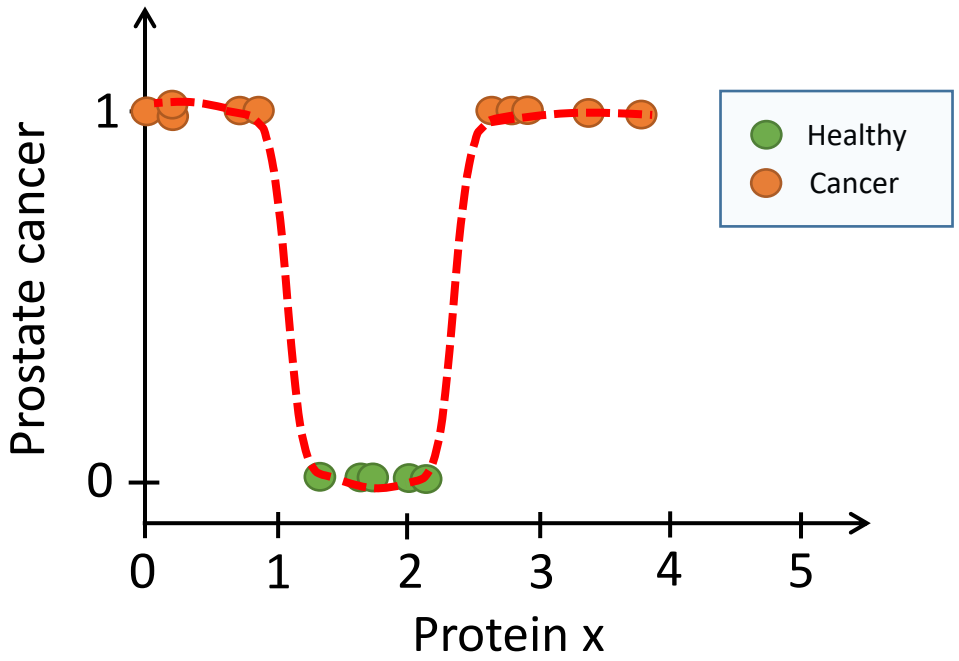
or

ANN without hidden layer



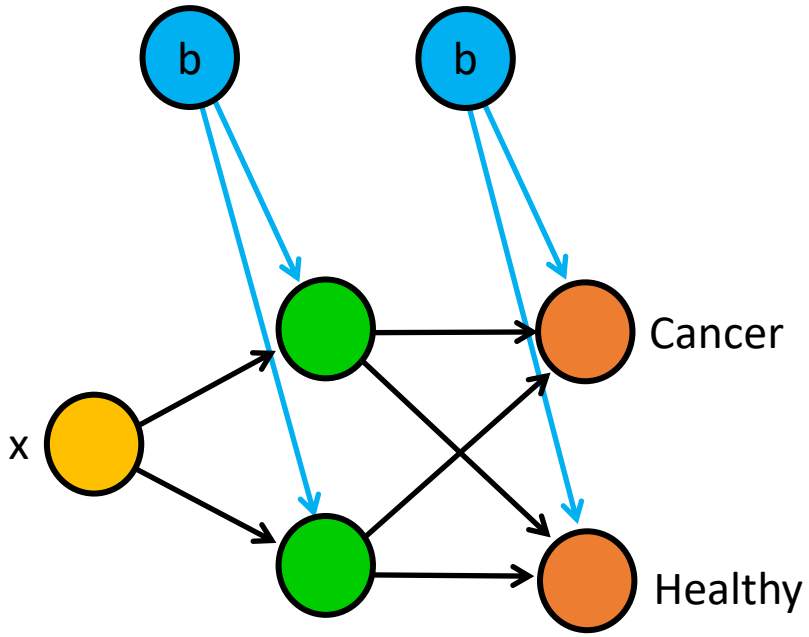
The hidden layer

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1



The hidden layer

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

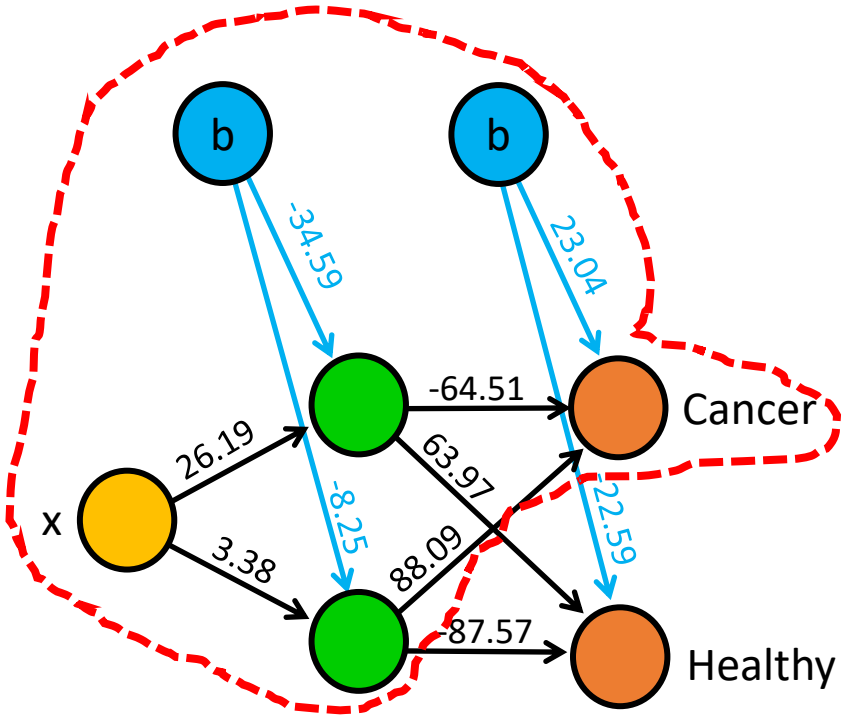


$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1



$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

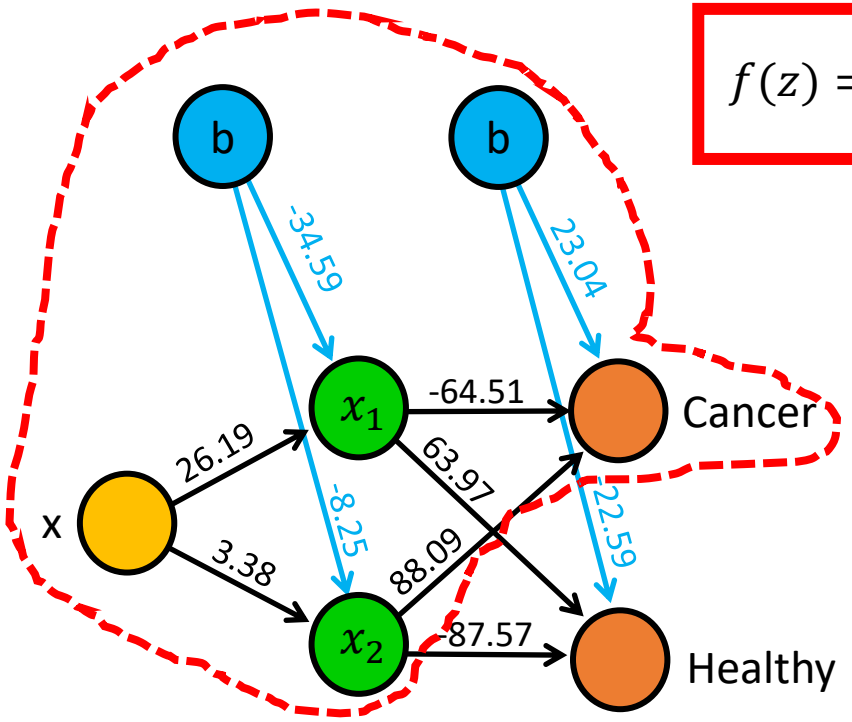
$$f(z) = \frac{1}{1 + e^{-z}}$$

The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

$$z = 23.04 - 64.51 \cdot x_1 + 88.09 \cdot x_2$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

The math

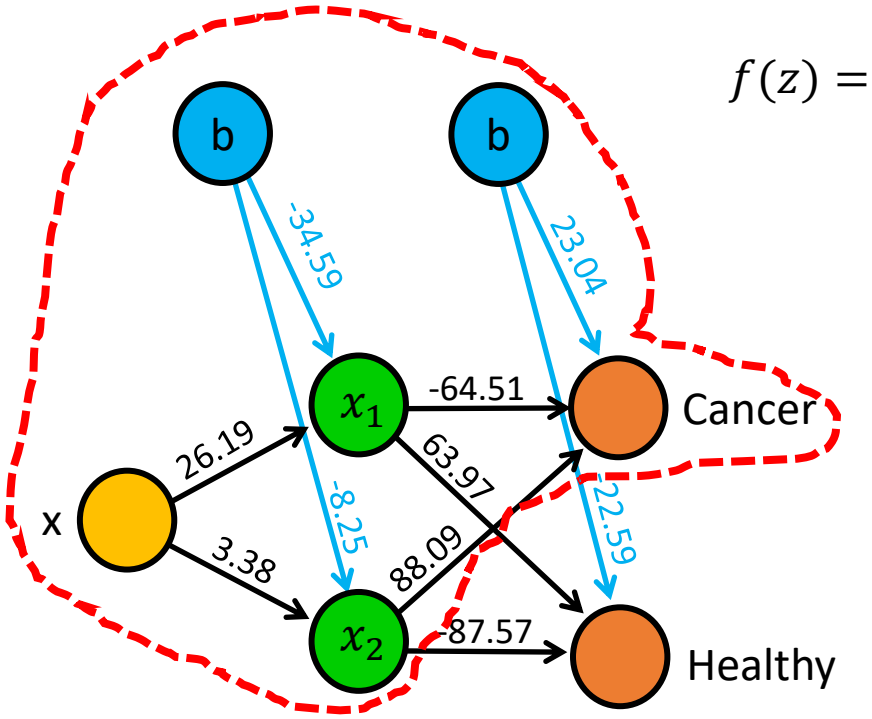
Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

$$z = 23.04 - 64.51 \cdot x_1 + 88.09 \cdot x_2$$

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

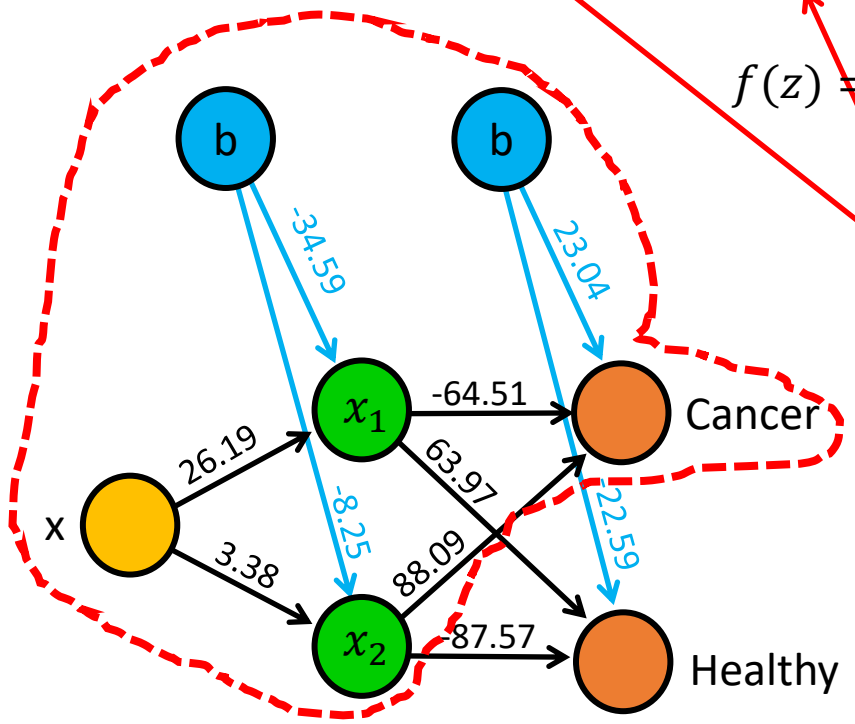


$$x_1 = \frac{1}{1 + e^{-(-34.59+26.59x)}}$$

$$x_2 = \frac{1}{1 + e^{-(-8.25+3.38x)}}$$

The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1



$$z = 23.04 - 64.51 \cdot x_1 + 88.09 \cdot x_2$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

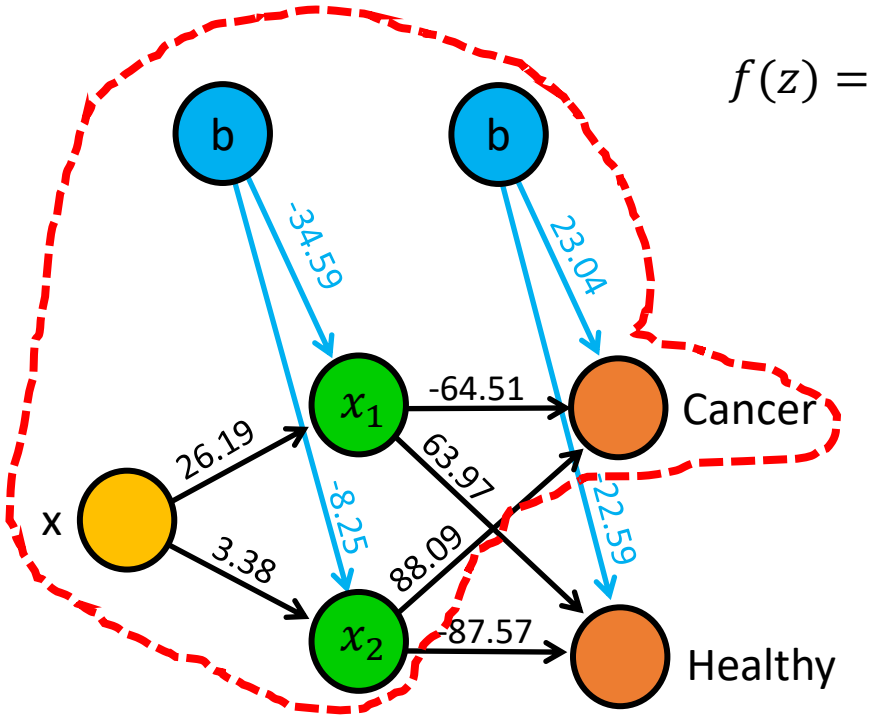
$$x_1 = \frac{1}{1 + e^{-(-34.59 + 26.59x)}}$$

$$x_2 = \frac{1}{1 + e^{-(-8.25 + 3.38x)}}$$

The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

$$z = 23.04 - 64.51 \cdot \frac{1}{1 + e^{-(-34.59+26.59x)}} + 88.09 \cdot \frac{1}{1 + e^{-(-8.25+3.38x)}}$$



$$f(z) = \frac{1}{1 + e^{-z}}$$

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

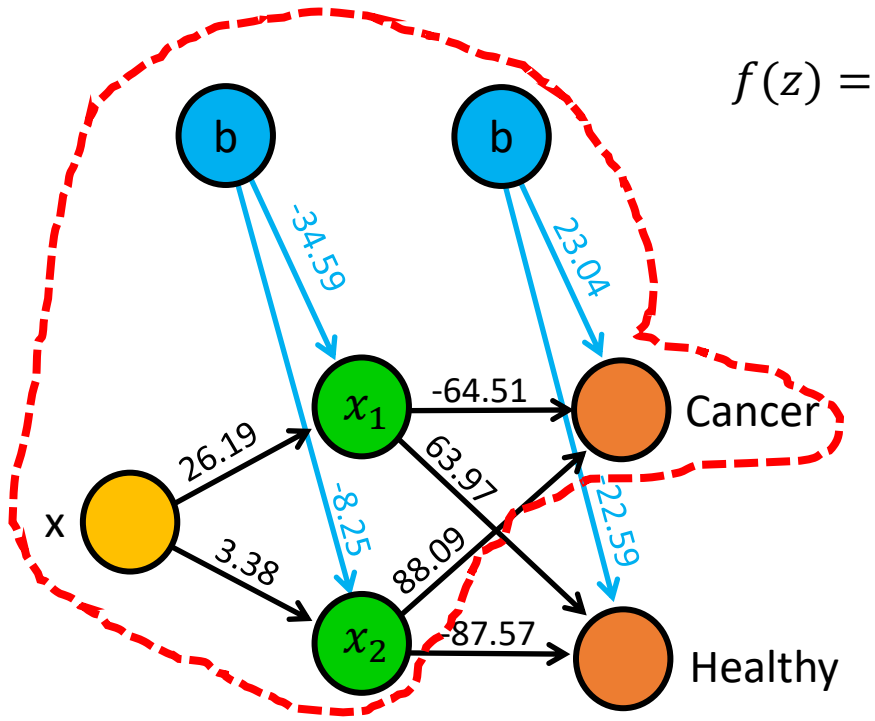
The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

$$z = 23.04 - 64.51 \cdot \frac{1}{1 + e^{-(-34.59+26.59x)}} + 88.09 \cdot \frac{1}{1 + e^{-(-8.25+3.38x)}}$$

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$



$$f(z) = \frac{1}{1 + e^{-z}}$$

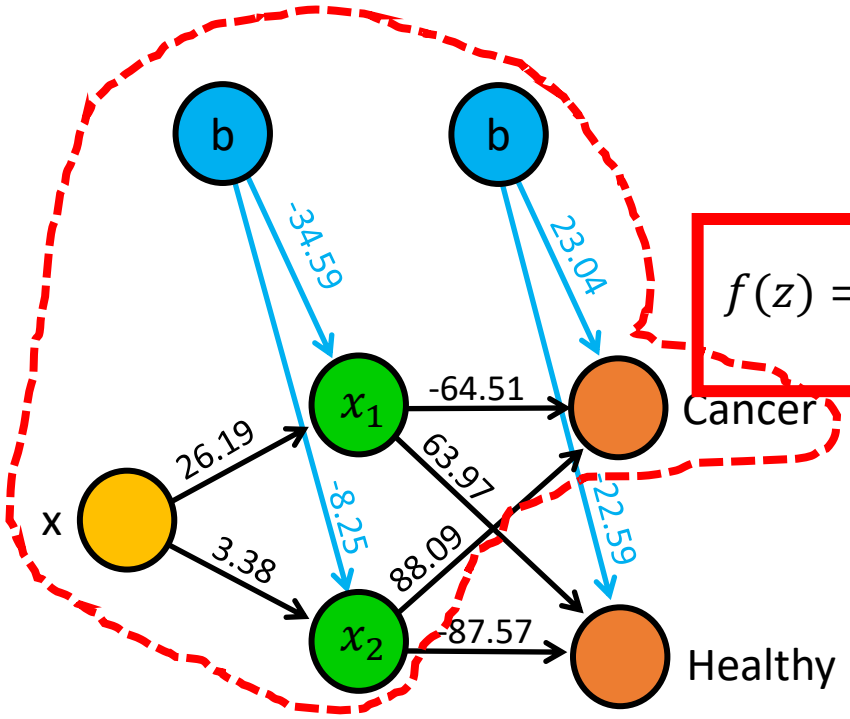
The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

$$z = 23.04 - 64.51 \cdot \frac{1}{1 + e^{-(-34.59+26.59x)}} + 88.09 \cdot \frac{1}{1 + e^{-(-8.25+3.38x)}}$$

$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$f(z) = \frac{1}{1 + e^{-z}}$$

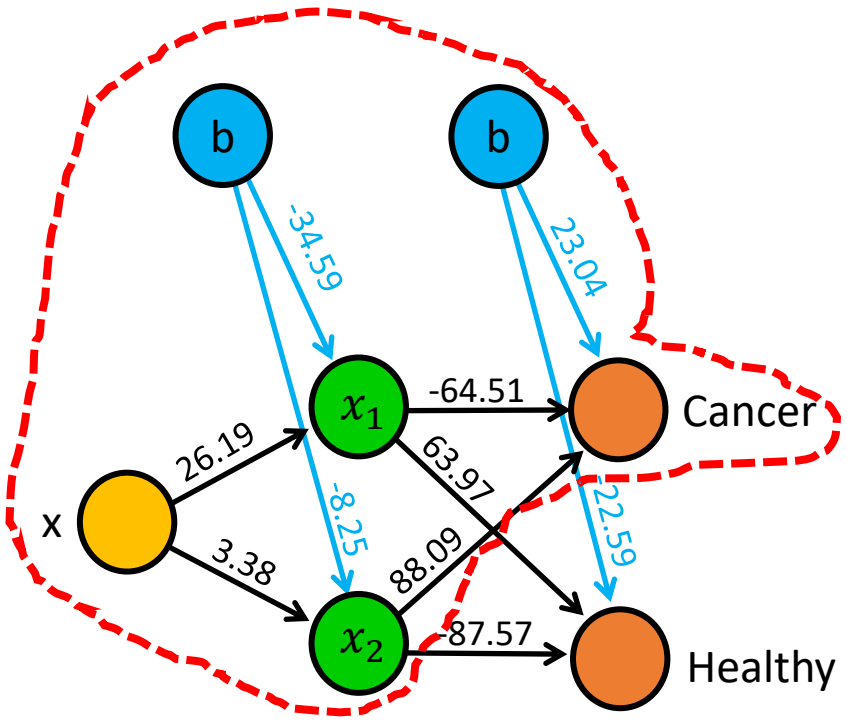


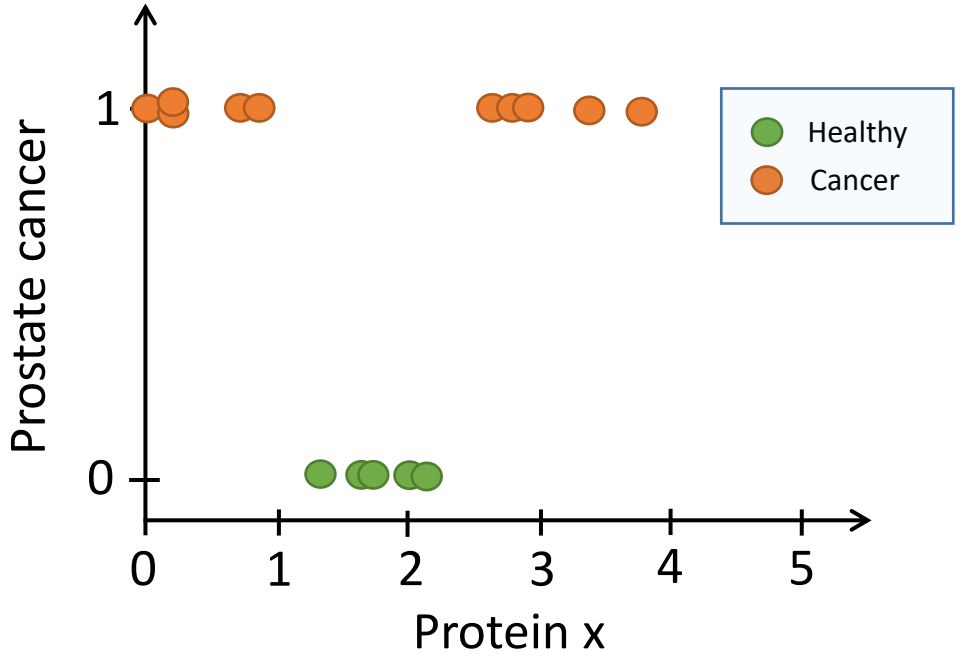
$$f(z) = \frac{1}{1 + e^{-(23.04 - 64.51 \cdot \frac{1}{1+e^{-(-34.59+26.59x)}} + 88.09 \cdot \frac{1}{1+e^{-(-8.25+3.38x)}})}}$$

The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

$$f(z) = \frac{1}{1 + e^{-(23.04 - 64.51 \cdot \frac{1}{1+e^{-(-34.59+26.19x)}} - 88.09 \cdot \frac{1}{1+e^{-(-8.25+3.38x)}})}}$$

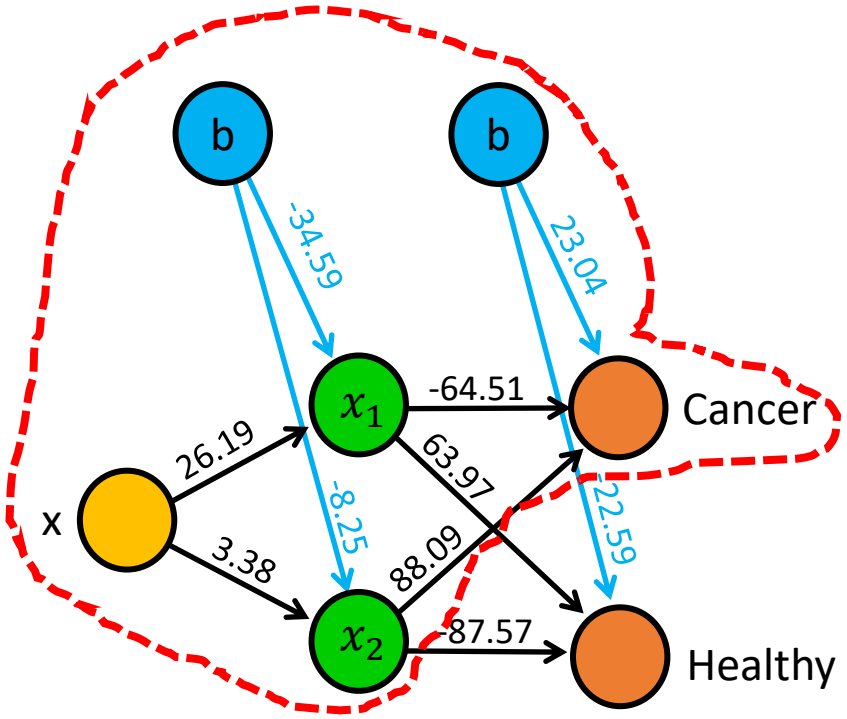


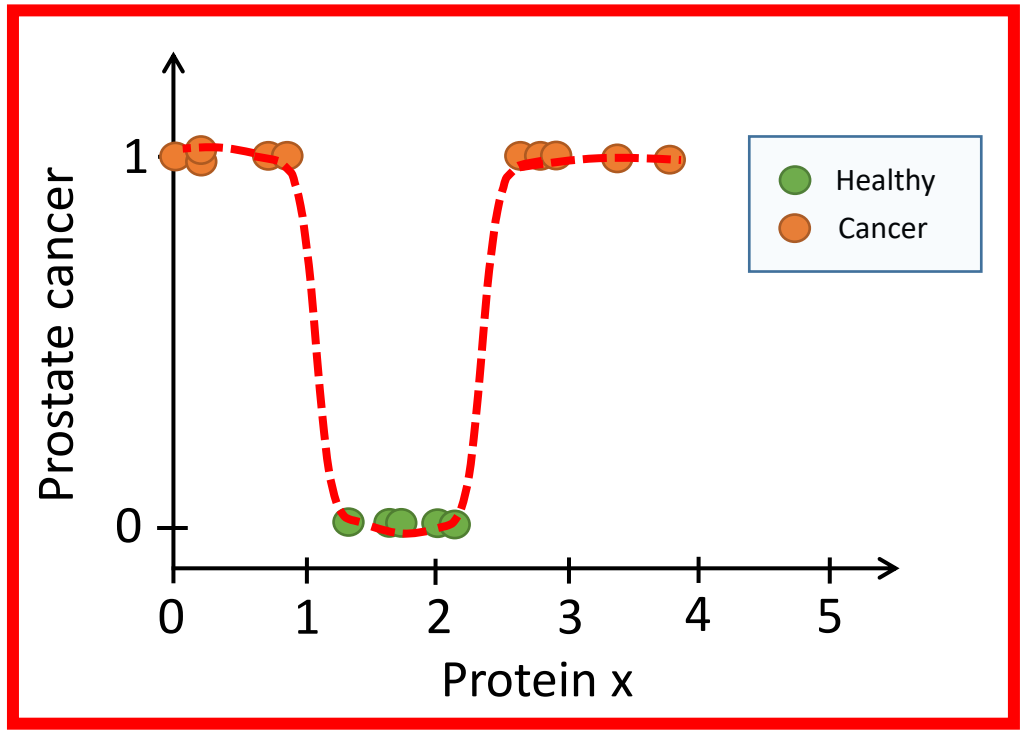


The math

Status	x
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Healthy	2.1
Healthy	2.0
Healthy	1.7
Healthy	1.6
Healthy	1.4
Cancer	0.9
Cancer	0.8
Cancer	0.2
Cancer	0.2
Cancer	0.1

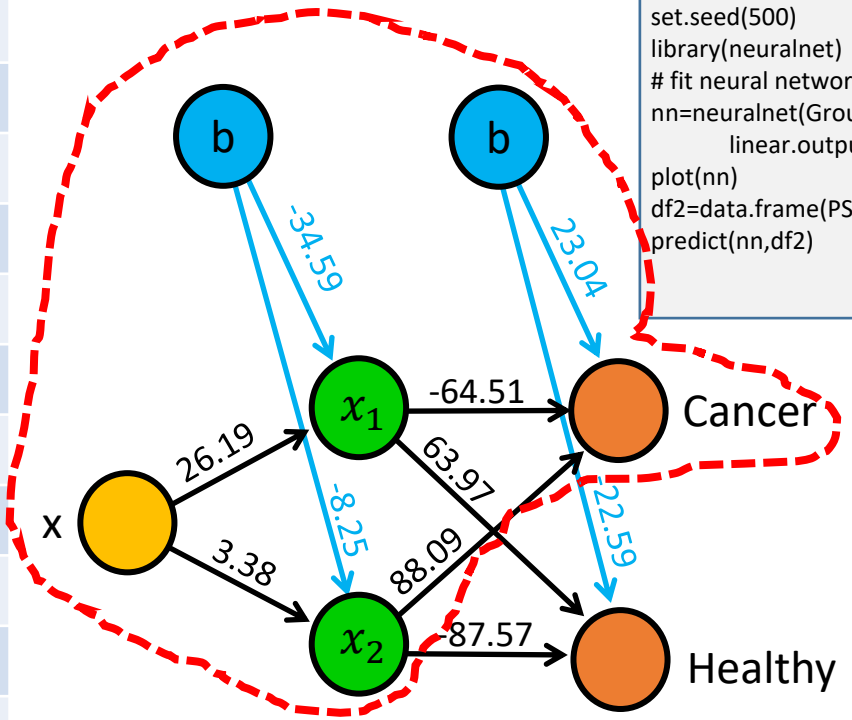
$$f(z) = \frac{1}{1 + e^{-(23.04 - 64.51 \cdot \frac{1}{1+e^{-(-34.59+26.59x)}} + 88.09 \cdot \frac{1}{1+e^{-(-8.25+3.38x)}})}}$$



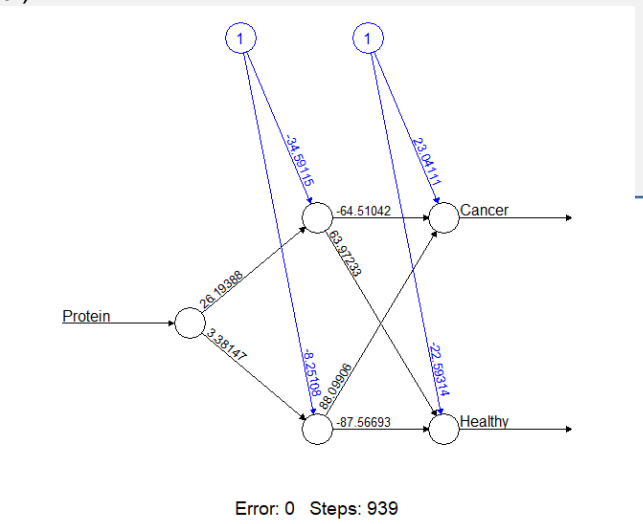


R code with hidden nodes

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8

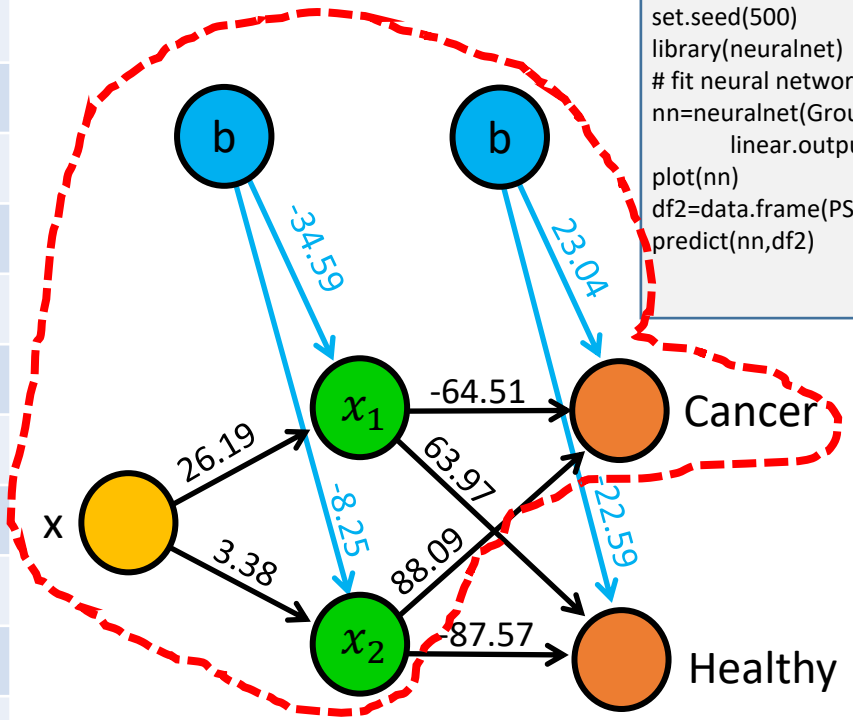



```
rm(list=ls())
Protein<-c(3.8, 3.4, 2.9, 2.8, 2.7,    2.1, 2.0, 1.7, 1.6, 1.4,    0.9, 0.8, 0.2,0.2,0.1) # PSA
Group<-factor(rep(c("Cancer","Healthy","Cancer"),c(5,5,5)),levels=c("Healthy","Cancer"))
df<-data.frame(Group,Protein)

state=ifelse(Group=="Cancer",1,0) # Recode to Cancer -> 1, Healthy -> 0
plot(Protein,state,col=Group,xlab="PSA µg/L",ylab="Probability",xlim=c(0,5))
legend("bottomright",levels(Group),col=1:2,pch=1)

set.seed(500)
library(neuralnet)
# fit neural network
nn=neuralnet(Group~Protein,data=df, hidden=c(2),act.fct = "logistic",
             linear.output = FALSE,err.fct="ce",threshold = 1e-8)
plot(nn)
df2=data.frame(PSA=0.5)
predict(nn,df2)
```



Try many different initial values

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8

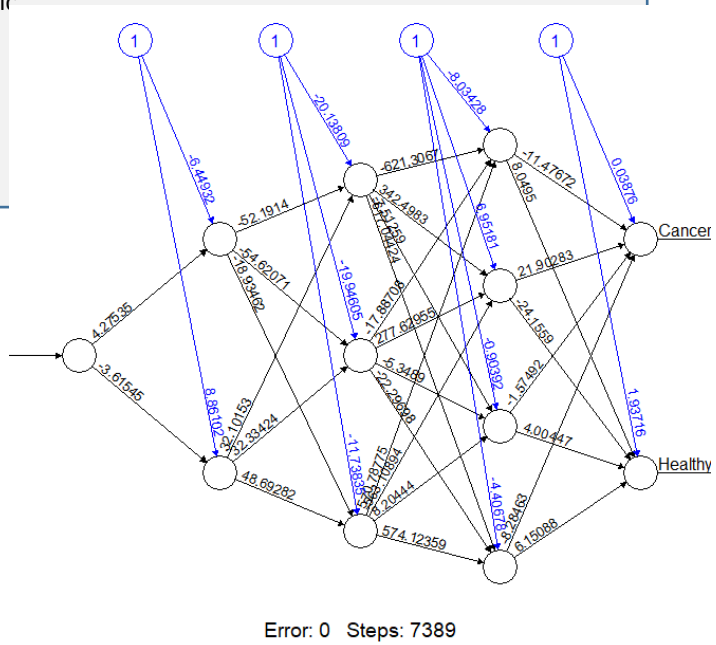



```
rm(list=ls())
Protein<-c(3.8, 3.4, 2.9, 2.8, 2.7,    2.1, 2.0, 1.7, 1.6, 1.4,    0.9, 0.8, 0.2,0.2,0.1) # PSA
Group<-factor(rep(c("Cancer","Healthy","Cancer"),c(5,5,5)),levels=c("Healthy","Cancer"))
df<-data.frame(Group,Protein)

state=ifelse(Group=="Cancer",1,0) # Recode to Cancer -> 1, Healthy -> 0
plot(Protein,state,col=Group,xlab="PSA µg/L",ylab="Probability",xlim=c(0,5))
legend("bottomright",levels(Group),col=1:2,pch=1)

set.seed(500)
library(neuralnet)
# fit neural network
nn=neuralnet(Group~Protein,data=df, hidden=c(2),act.fct = "logistic",
             linear.output = FALSE,err.fct="ce",threshold = 1e-8, rep=10)
plot(nn)
df2=data.frame(PSA=0.5)
predict(nn,df2)
```

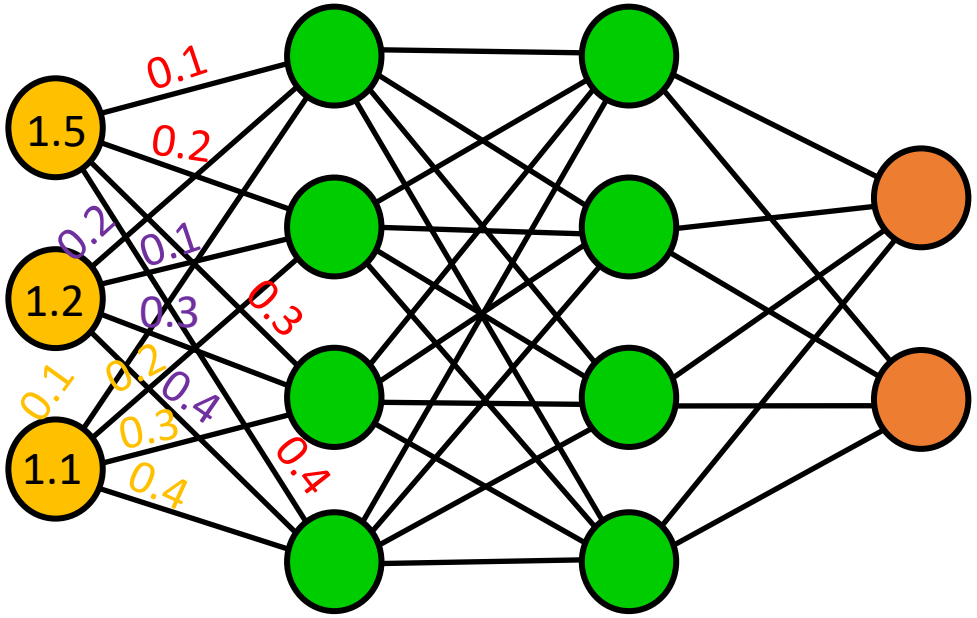
R code with several hidden nodes

Status	PSA
Cancer	3.8
Cancer	3.4
Cancer	2.9
Cancer	2.8
Cancer	2.7
Cancer	2.1
Cancer	1.6
Healthy	2.5
Healthy	2.0
Healthy	1.7
Healthy	1.4
Healthy	1.2
Healthy	0.9
Healthy	0.8

```
rm(list=ls())
Protein<-c(3.8, 3.4, 2.9, 2.8, 2.7,    2.1, 2.0, 1.7, 1.6, 1.4,    0.9, 0.8, 0.2,0.2,0.1) # PSA
Group<-factor(rep(c("Cancer","Healthy","Cancer"),c(5,5,5)),levels=c("Healthy","Cancer"))
df<-data.frame(Group,Protein)

state=ifelse(Group=="Cancer",1,0) # Recode to Cancer -> 1, Healthy -> 0
plot(Protein,state,col=Group,xlab="PSA µg/L",ylab="Probability",xlim=c(0,5))
legend("bottomright",levels(Group),col=1:2,pch=1)

set.seed(500)
library(neuralnet)
# fit neural network
nn=neuralnet(Group~Protein,data=df, hidden=c(2,3,3),act.fct = "logistic"
          linear.output = FALSE,err.fct="ce",threshold = 1e-8)
plot(nn)
df2=data.frame(PSA=0.5)
predict(nn,df2)
```



ANN = Matrix multiplication



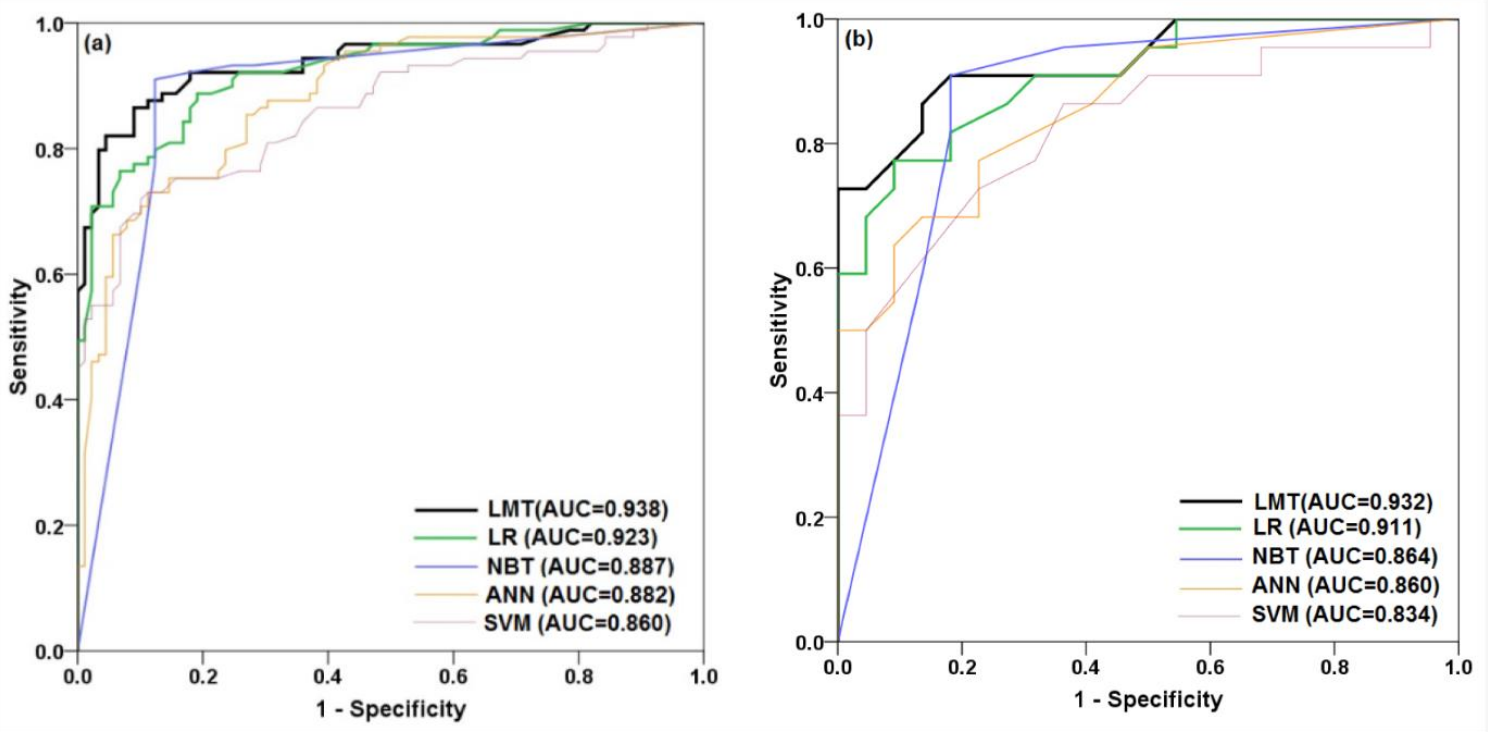
$$z = w_0 + \sum_{i=1}^{n} w_i \cdot x_i$$

$$\begin{matrix} & w_i & & & x_i \end{matrix}$$

$$\begin{bmatrix} 0.1 & 0.2 & 0.1 \\ 0.2 & 0.1 & 0.2 \\ 0.3 & 0.3 & 0.3 \\ 0.4 & 0.4 & 0.4 \end{bmatrix} \cdot \begin{bmatrix} 1.5 \\ 1.2 \\ 1.1 \end{bmatrix} + w_0$$

Logistic regression vs ANN

Figure 8. Receiver operating characteristic (ROC) curves and area under the receiver operatic characteristic curve (AUC) for the (**a**) training dataset and (**b**) validation dataset.

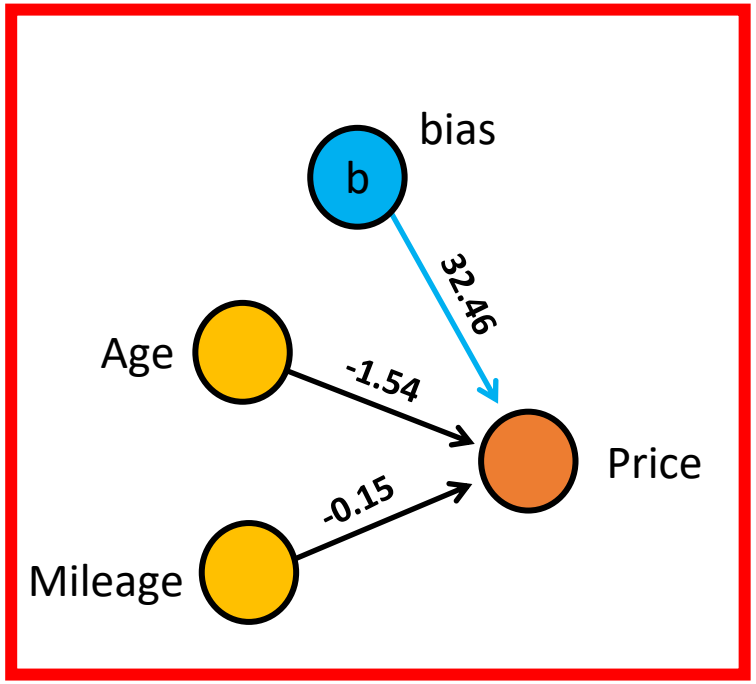


Hyperparameters in ANN

- Learning rate
- Batch Size
- Number of Epochs
- Number of Layers and Nodes
- Activation Functions
- Optimizer
- Initial Weights

ANN vs Linear regression

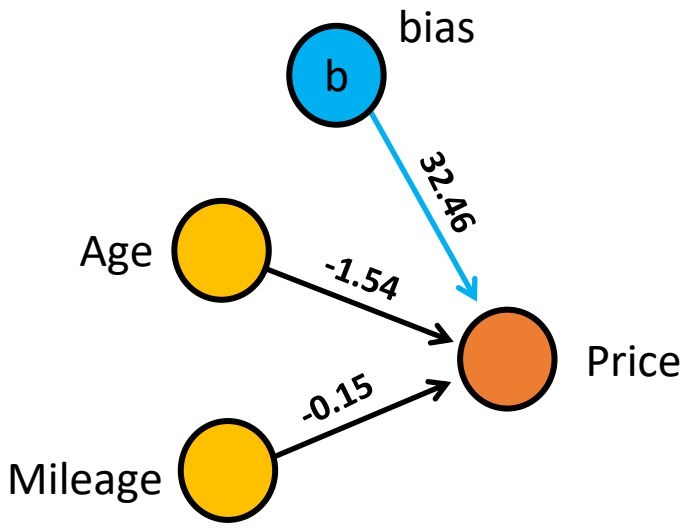
Car	Price (k€)	Age (years)	Mileage (k miles)
1	29	1	18
2	25	2	25
3	21	2	50
4	18	3	68
5	15	4	75
6	15	5	65



ANN vs Linear regression

Car	Price (k€)	Age (years)	Mileage (k miles)
1	29	1	18
2	25	2	25
3	21	2	50
4	18	3	68
5	15	4	75
6	15	5	65

$$z = w_0 + w_1 \cdot x_1 + w_2 \cdot x_2$$



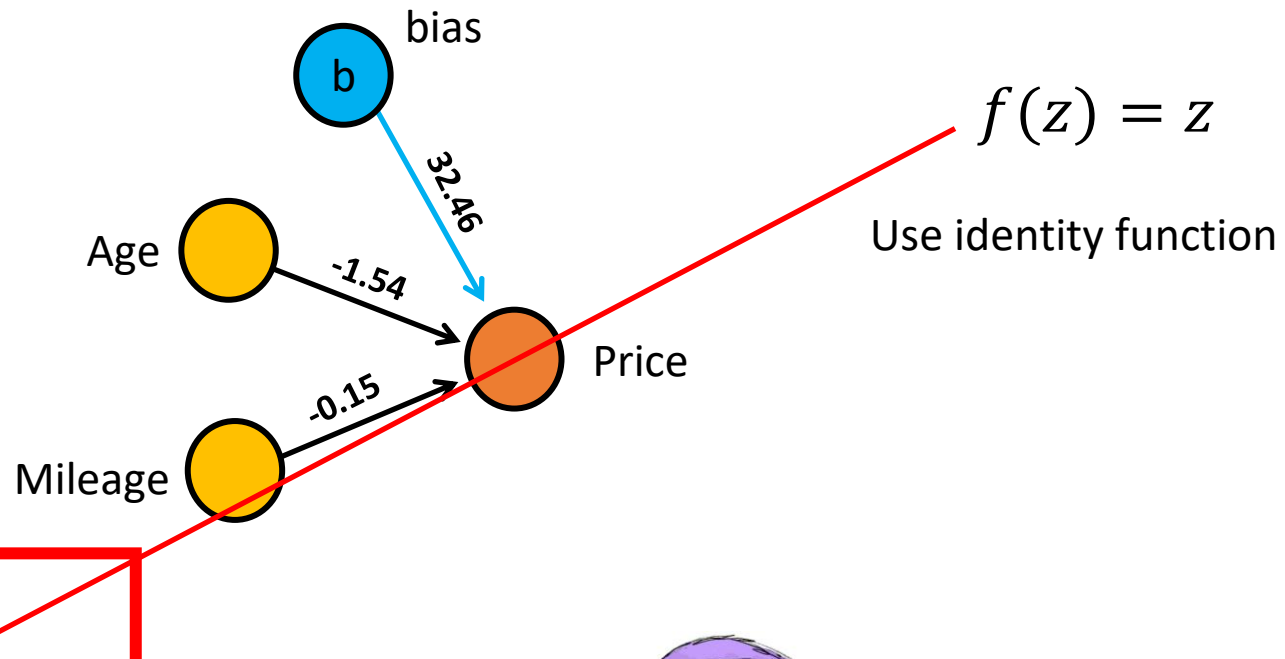
$$f(z) = z$$

Use identity function

ANN vs Linear regression

Car	Price (k€)	Age (years)	Mileage (k miles)
1	29	1	18
2	25	2	25
3	21	2	50
4	18	3	68
5	15	4	75
6	15	5	65

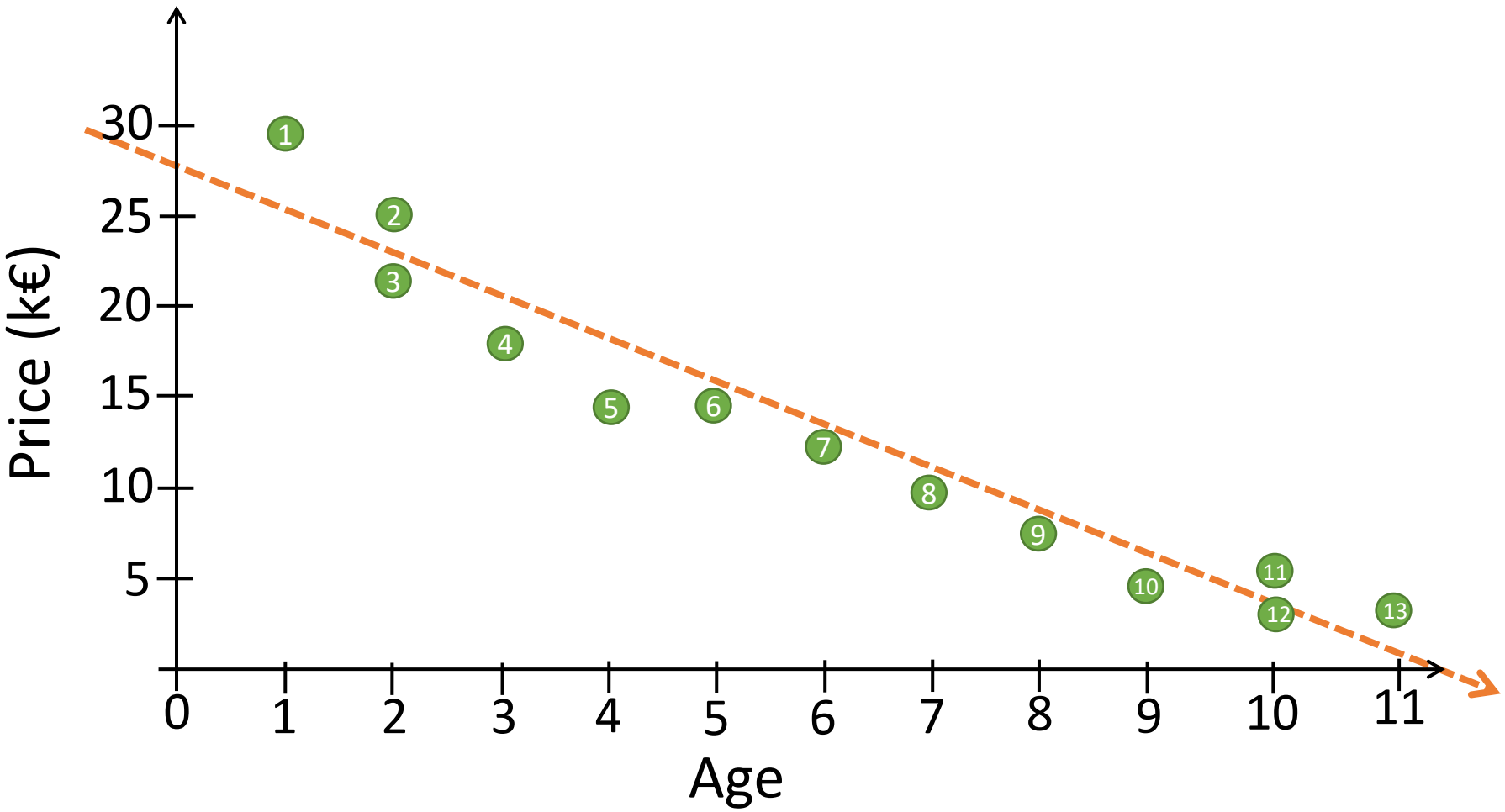
$$z = w_0 + w_1 \cdot x_1 + w_2 \cdot x_2$$



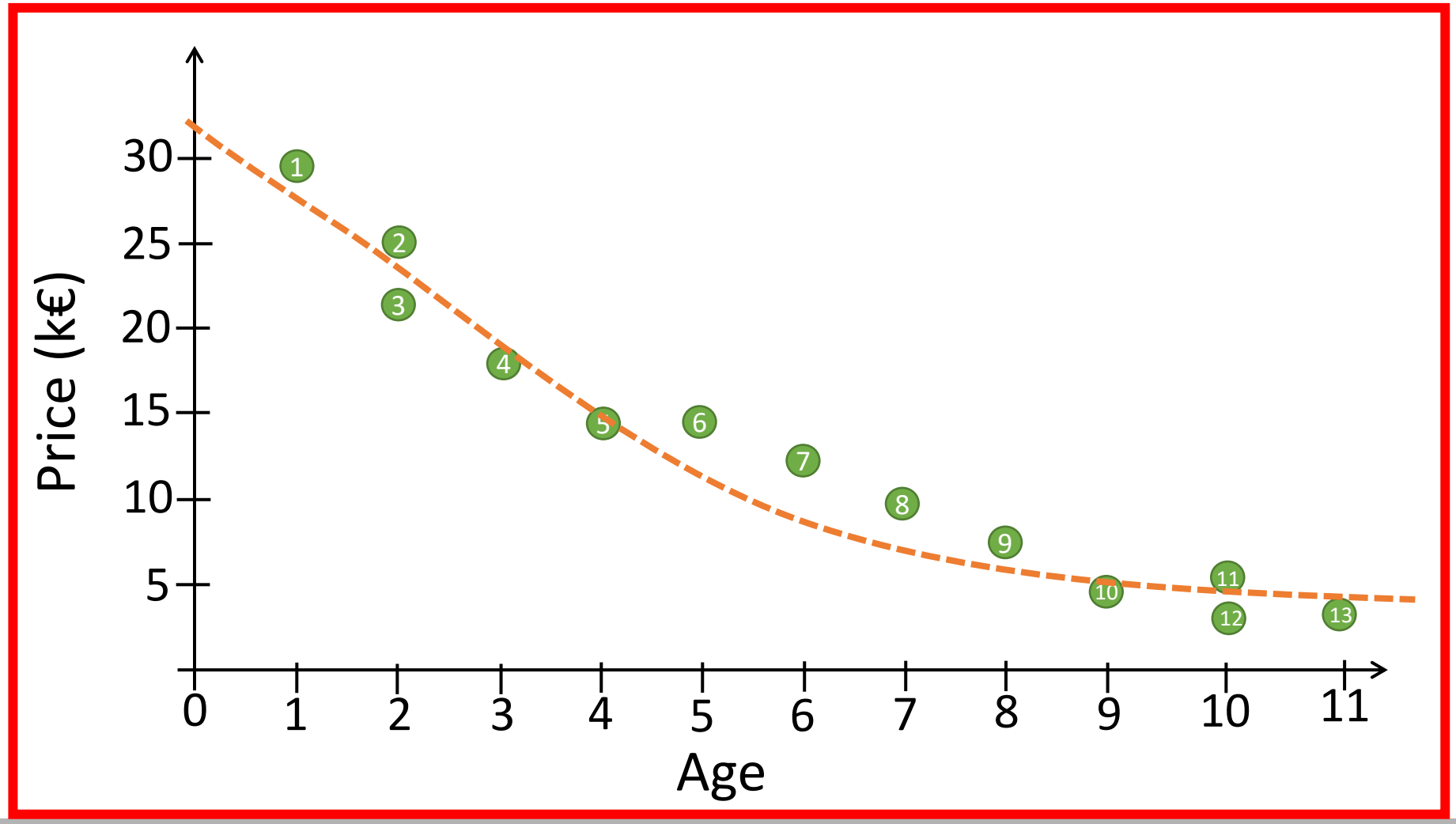
Multiple linear regression

$$\text{Price} = 32.46 - 1.54 \cdot \text{Age} - 0.15 \cdot \text{Mileage}$$

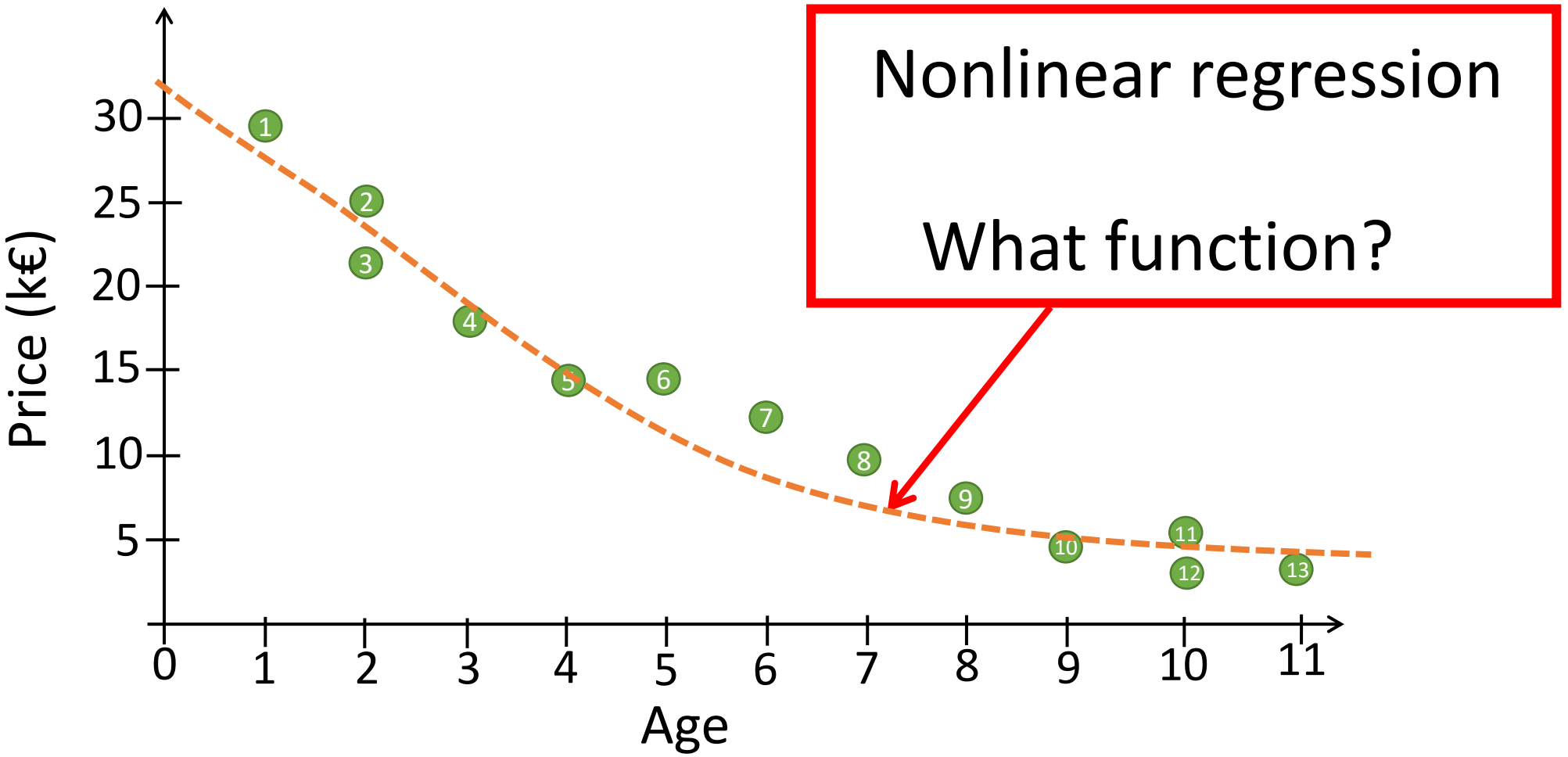
ANN vs Linear regression



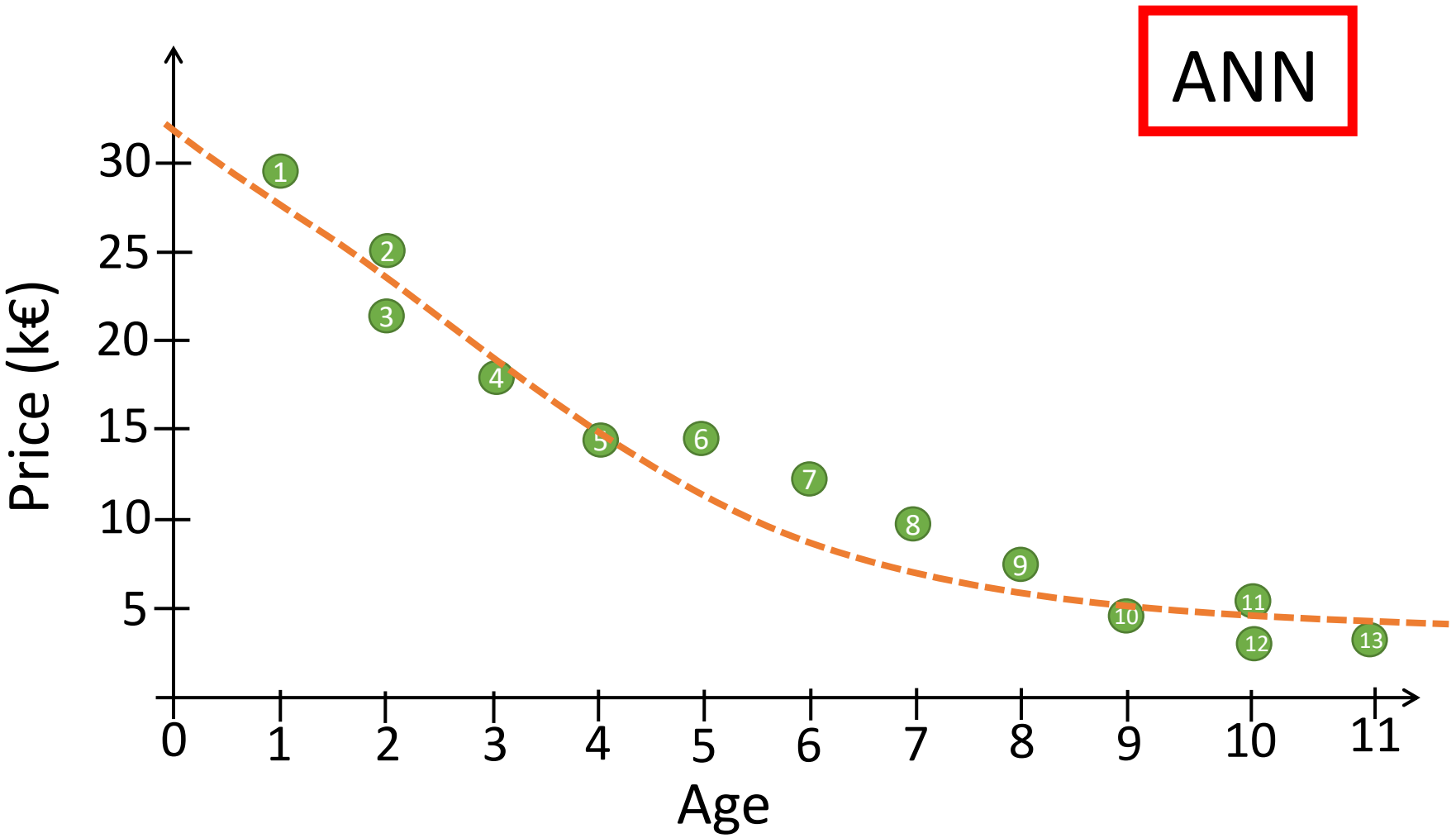
ANN vs Linear regression



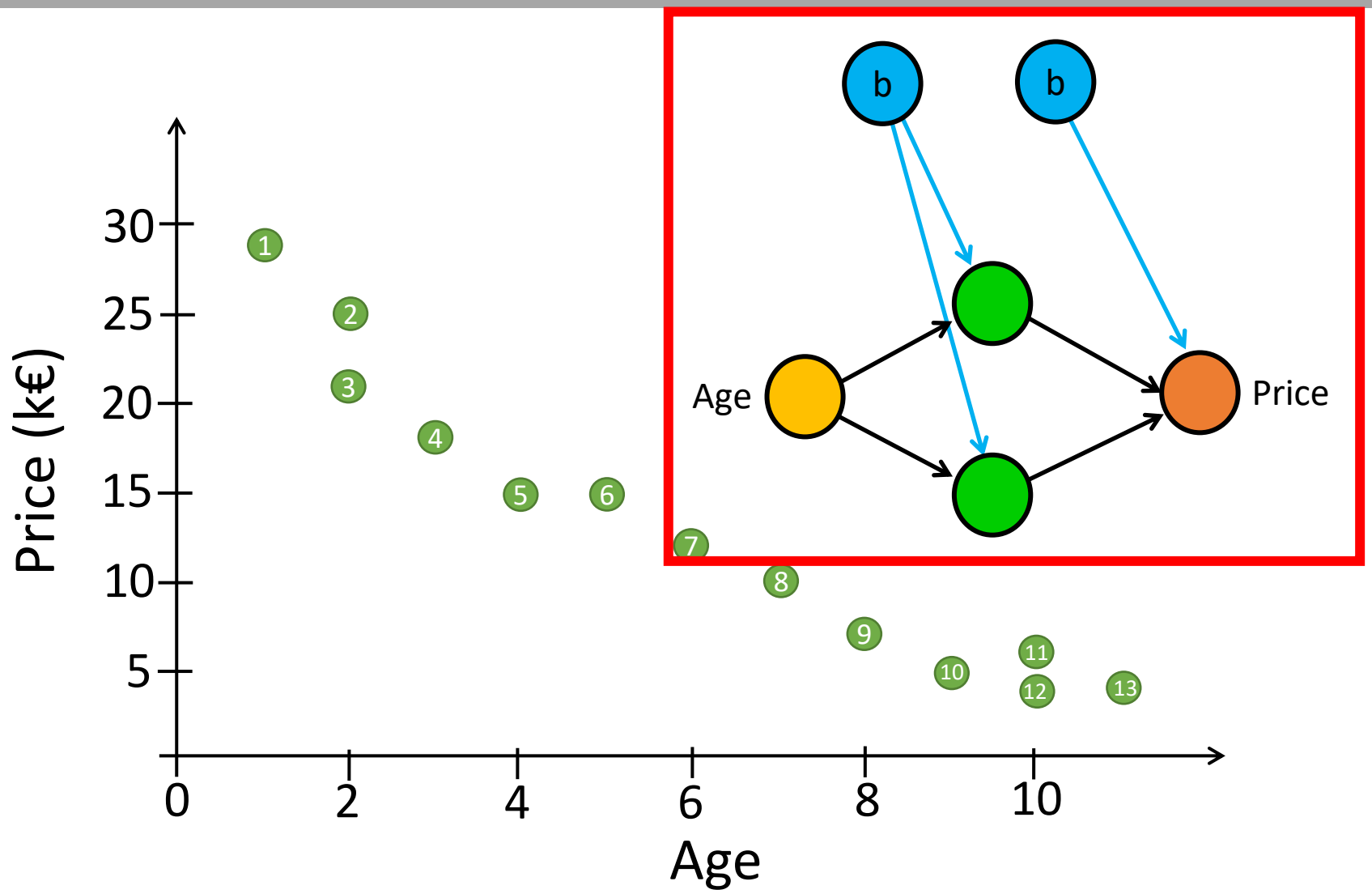
Nonlinear regression



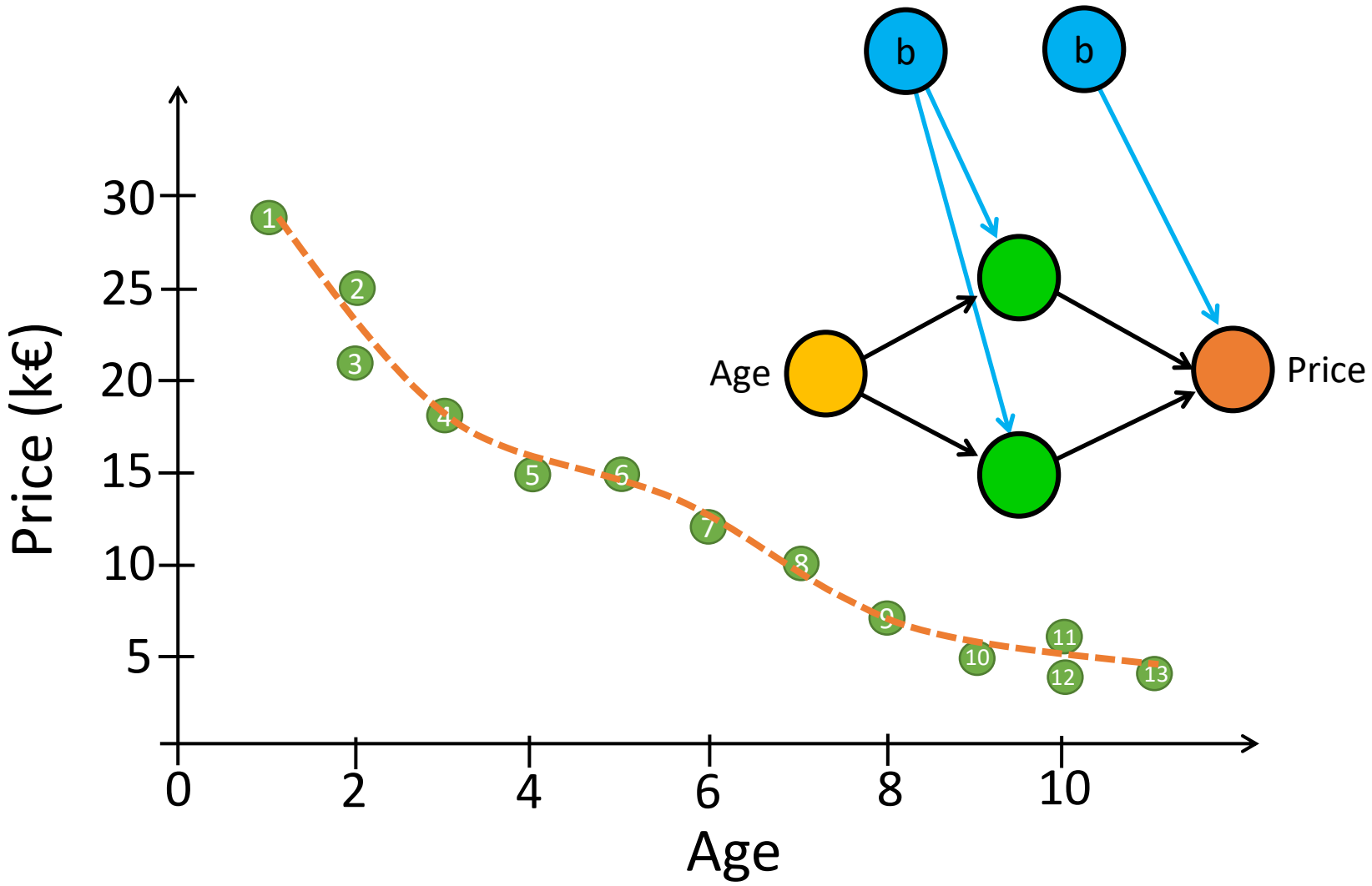
ANN


ANN
Price (k€)
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
9
10
11
Age
1
2
3
4
5
6
7
8
9
10
11
12
13

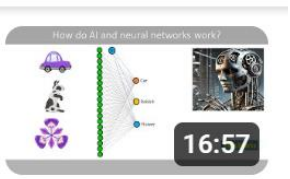
ANN

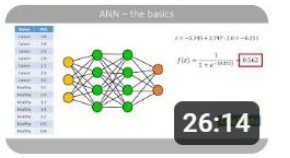

b
b
Age
Price
Price (k€)
30
25
20
15
10
5
0
2
4
6
8
10
Age
1
2
3
4
5
6
7
8
9
10
11
12
13

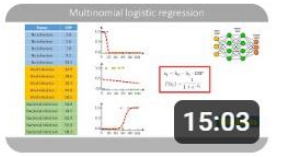
ANN

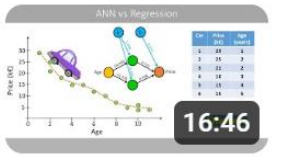

Price (k€)
30
25
20
15
10
5
0
2
4
6
8
10
Age
1
2
3
4
5
6
7
8
9
10
11
12
13
b
b
Age
Price

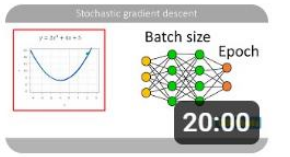
END

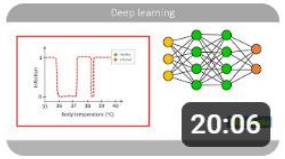

How do AI and neural networks work?
16:57

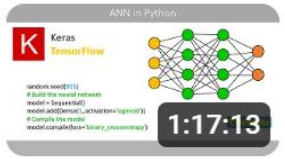

ANN – the basics
26:14

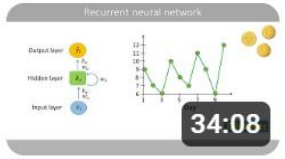

Multinomial logistic regression
15:03

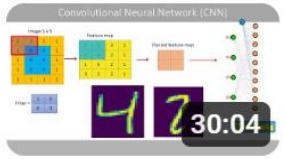

ANN vs Regression
16:46

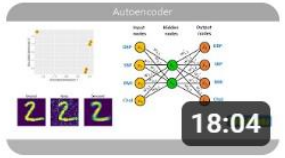

Stochastic gradient descent
Batch size
Epoch
20:00

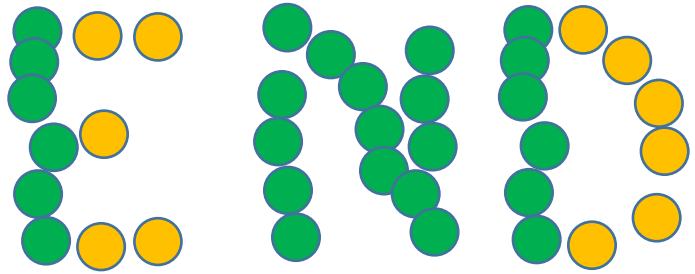

Deep learning
20:06


ANN in Python
Keras
TensorFlow
1:17:13


Recurrent neural network
34:08


Convolutional Neural Network (CNN)
30:04


Autoencoder
18:04


END