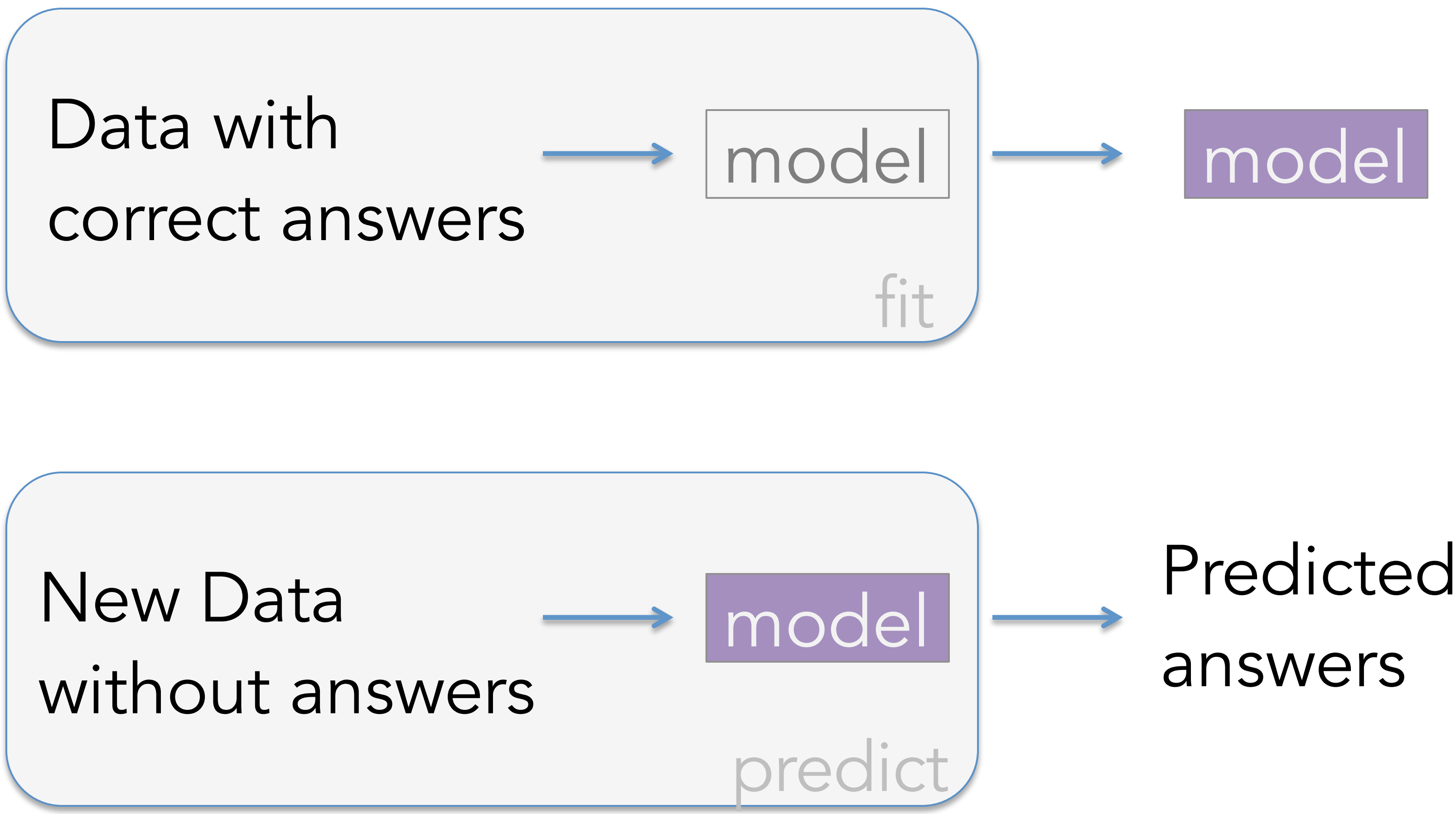
Supervised Learning

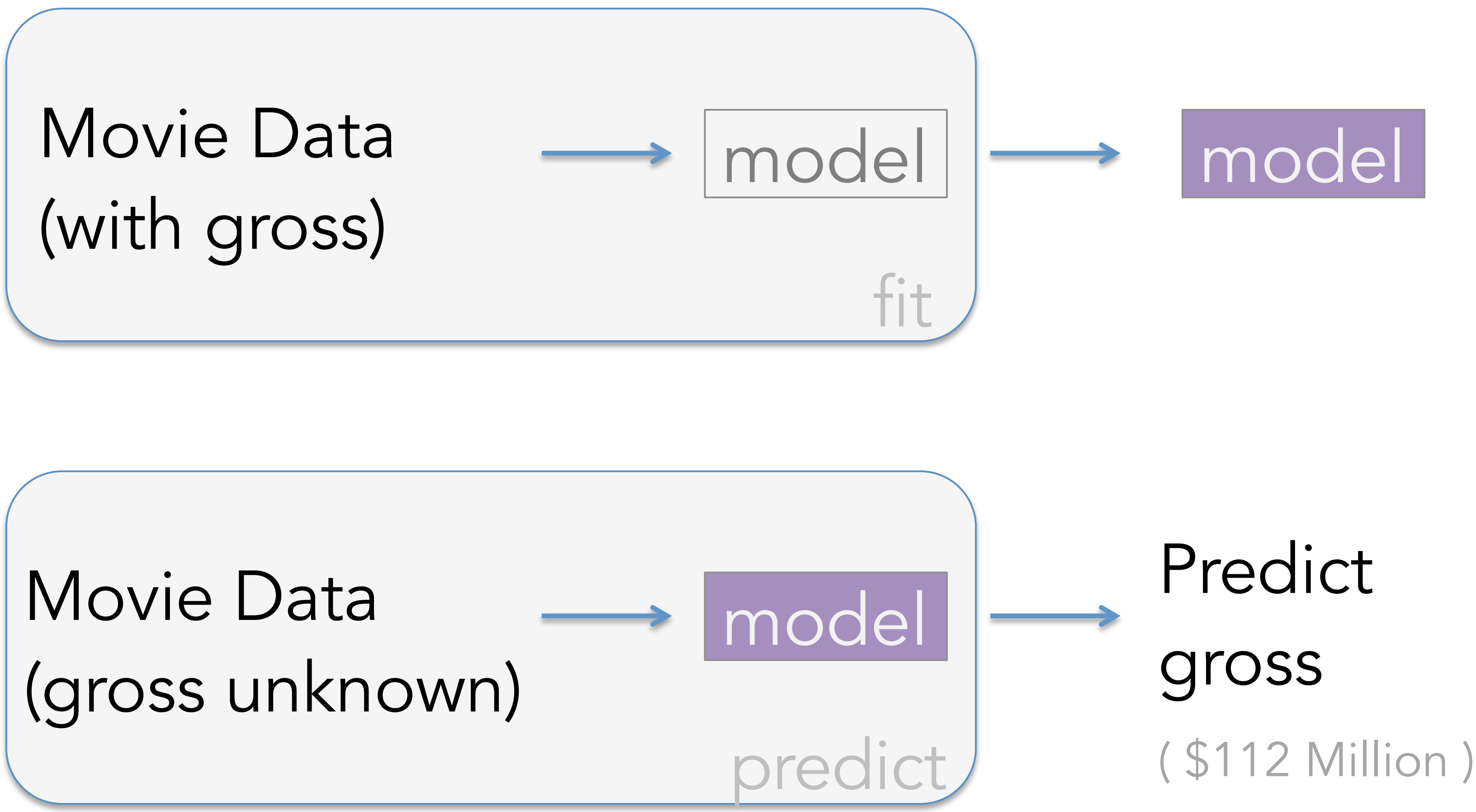
METIS

datascope

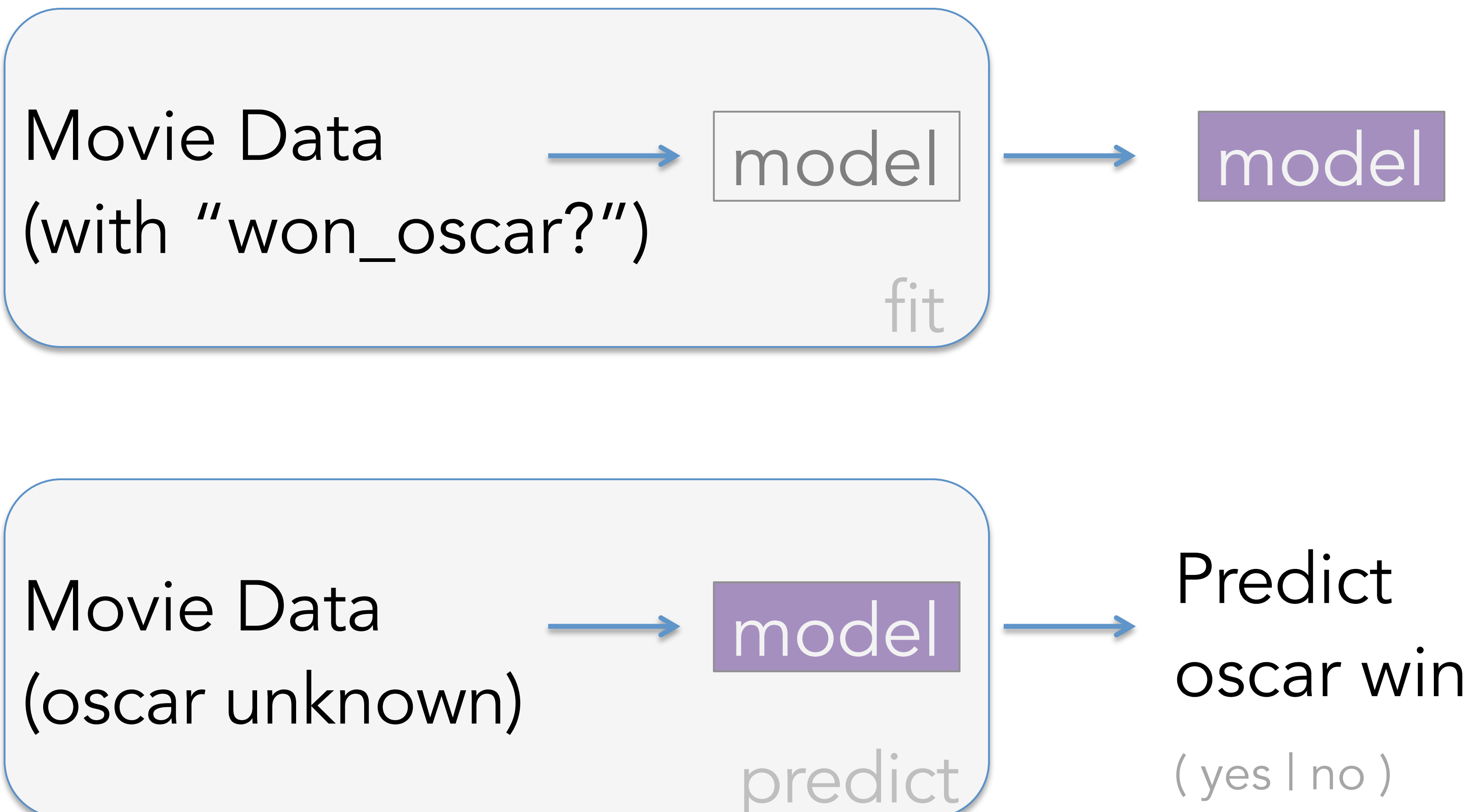
What is Supervised Learning?



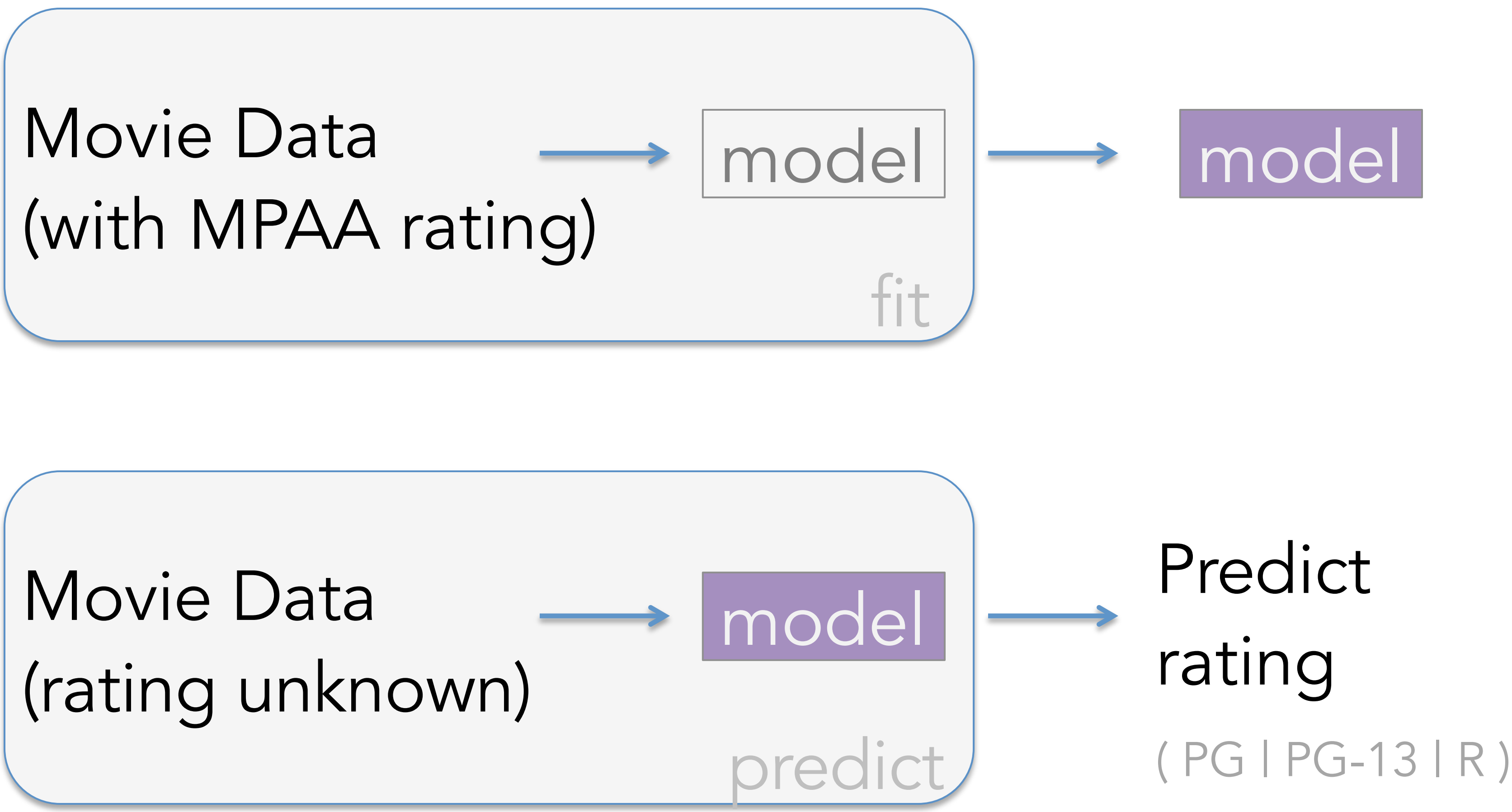
Regression:
"answers" are numeric



Classification:
"answers" are categories



Classification:
"answers" are categories



Classification: "answers" are categories

Breast cancer surgery data (including survival) → model → model

fit

Breast cancer surgery data (survival unknown) → model → Predict survival

predict

(survived | lost)

Classification:
"answers" are categories

Color, shape, weight, sweetness for a bunch of apples, bananas, peaches → model → model

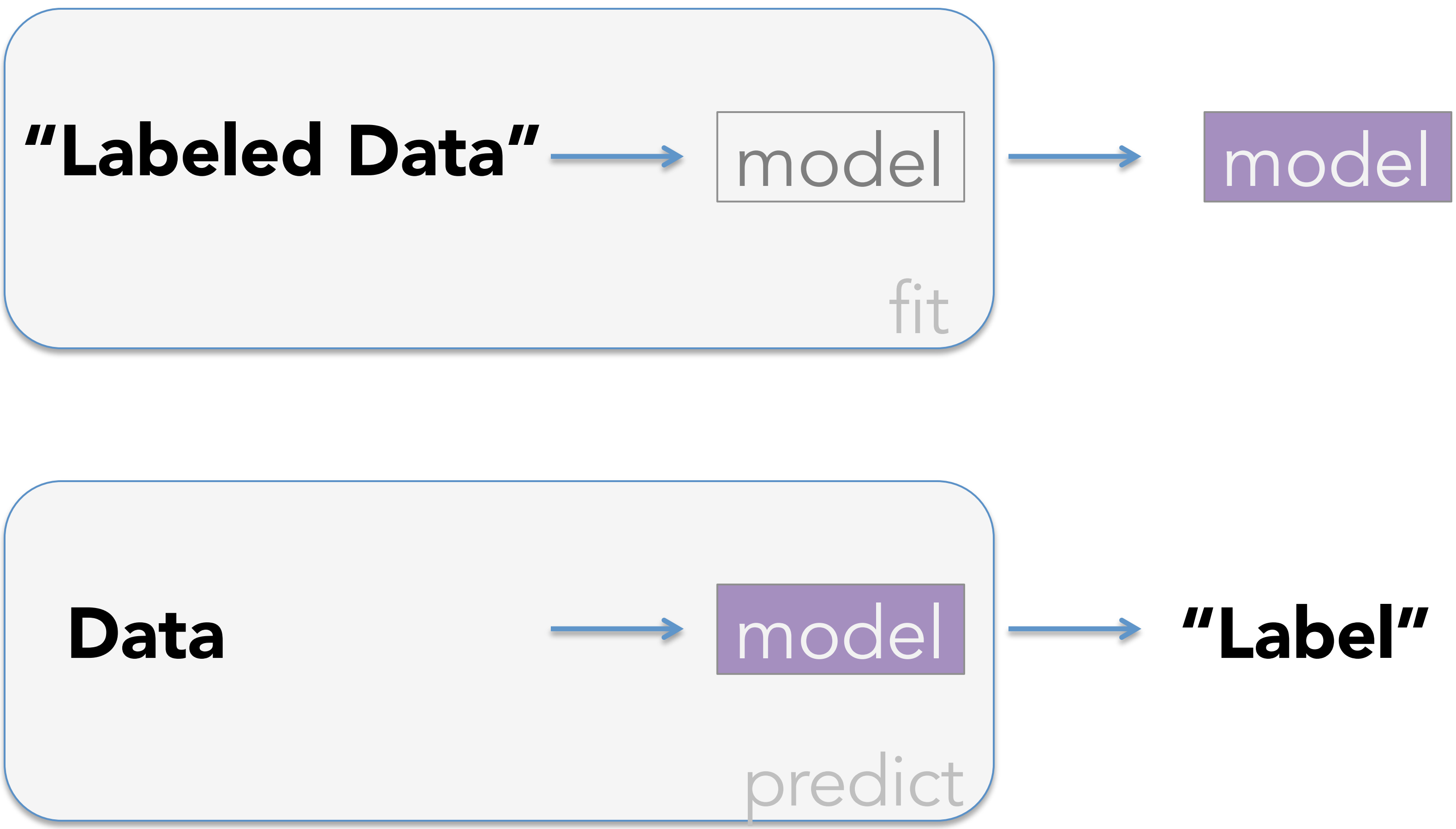
fit

Color, shape, weight, sweetness data → model → Predict fruit

predict

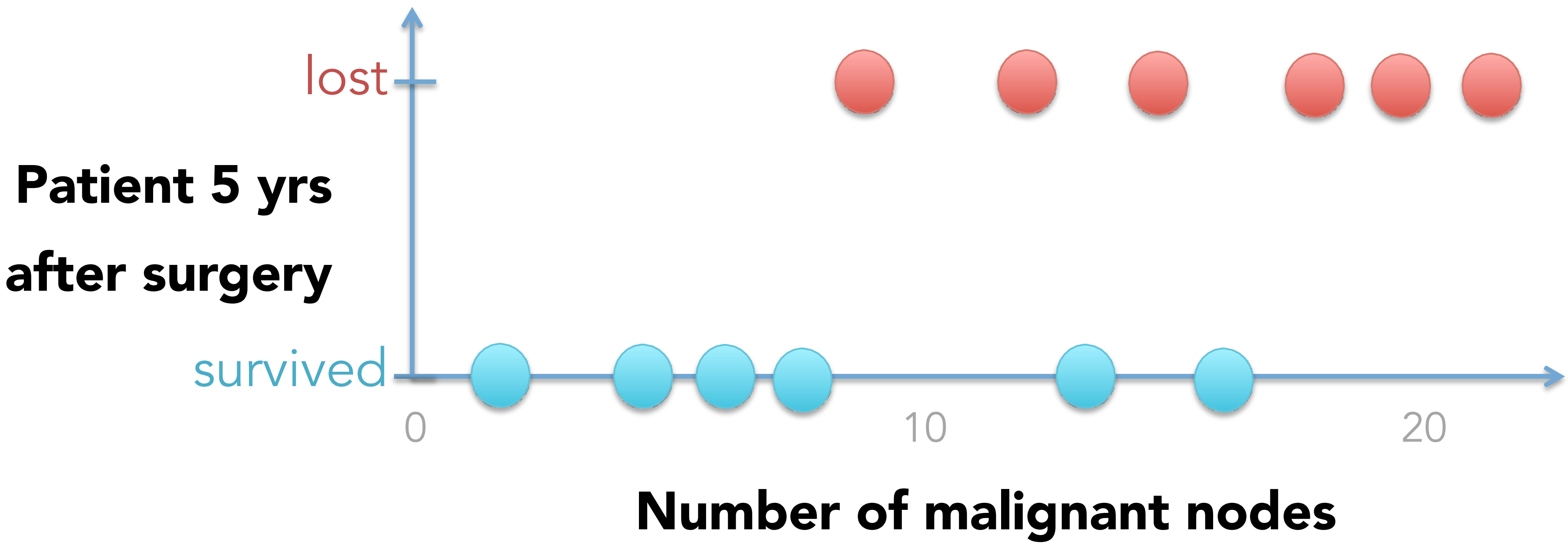
(apple | banana | peach)

Classification:
"answers" are categories

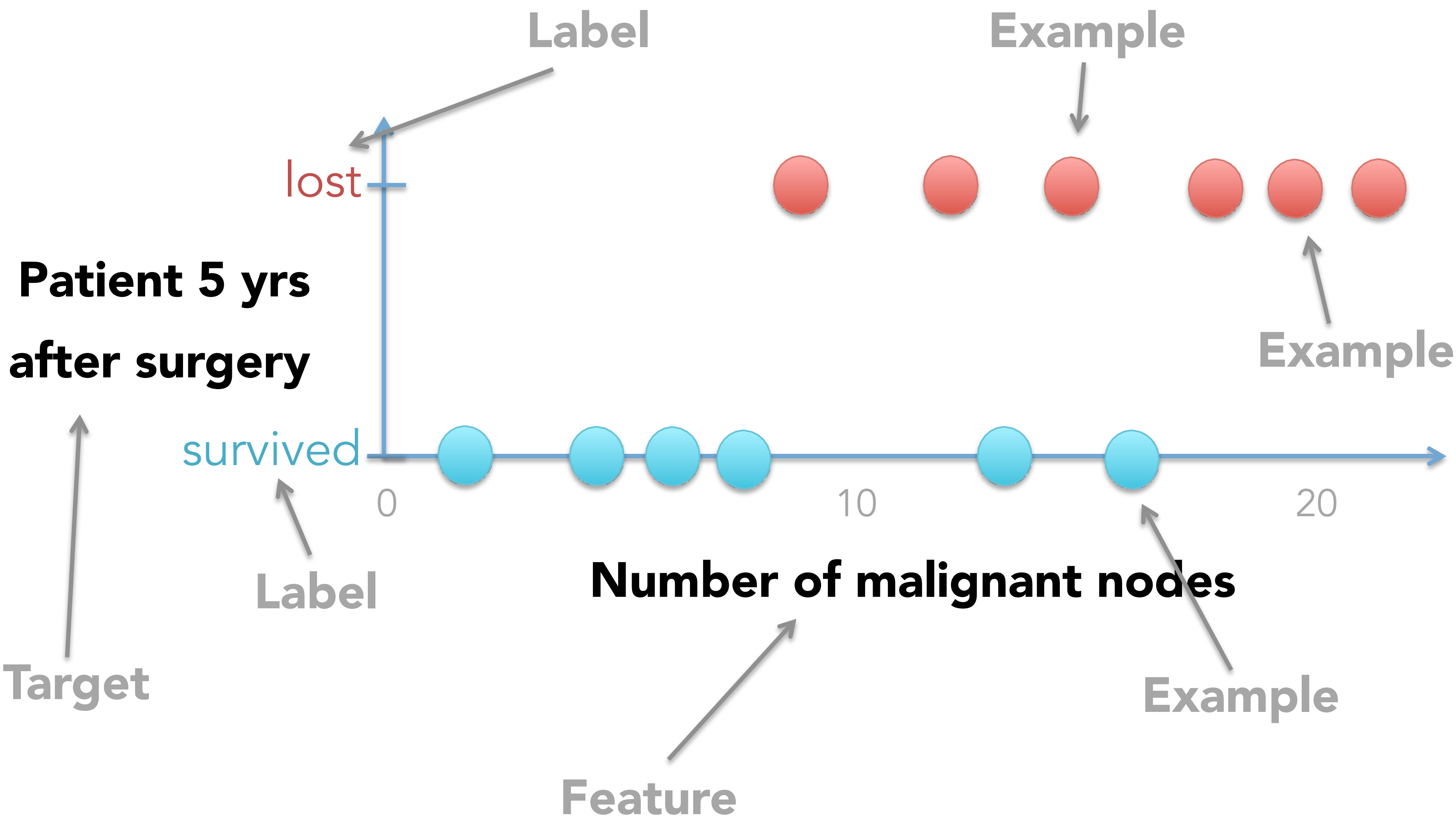


Example	each data point (one row)
Target	predicted property (column to predict)
Label	Target / category of the point (value of target column)
Feature	a property of the point used for prediction (non-target columns in the model)

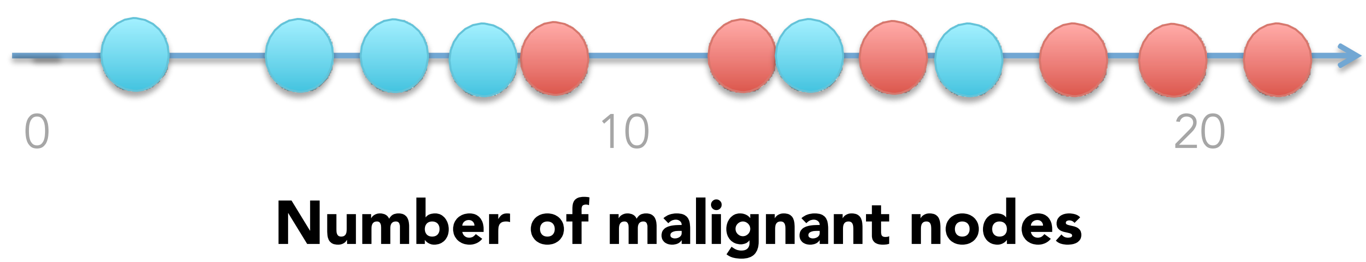
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



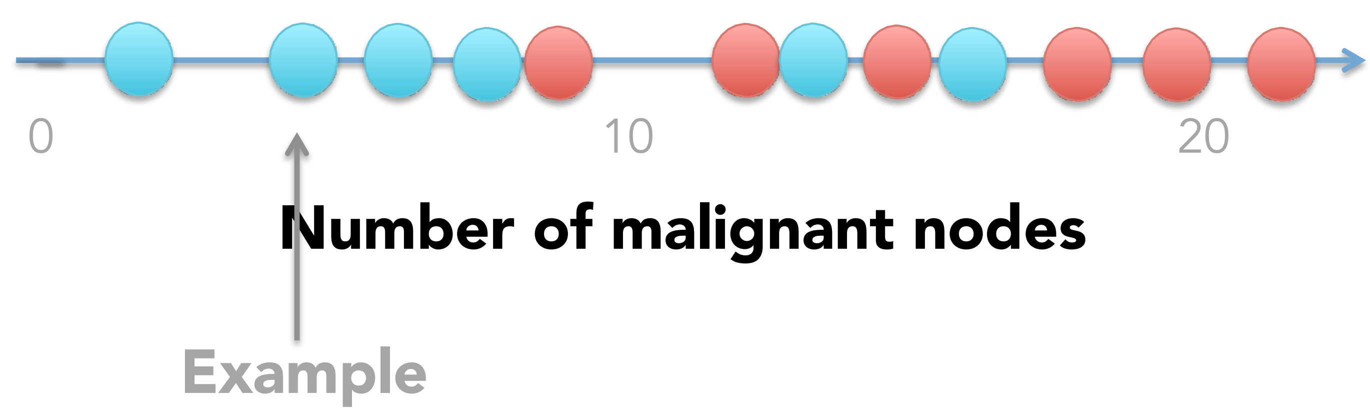
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



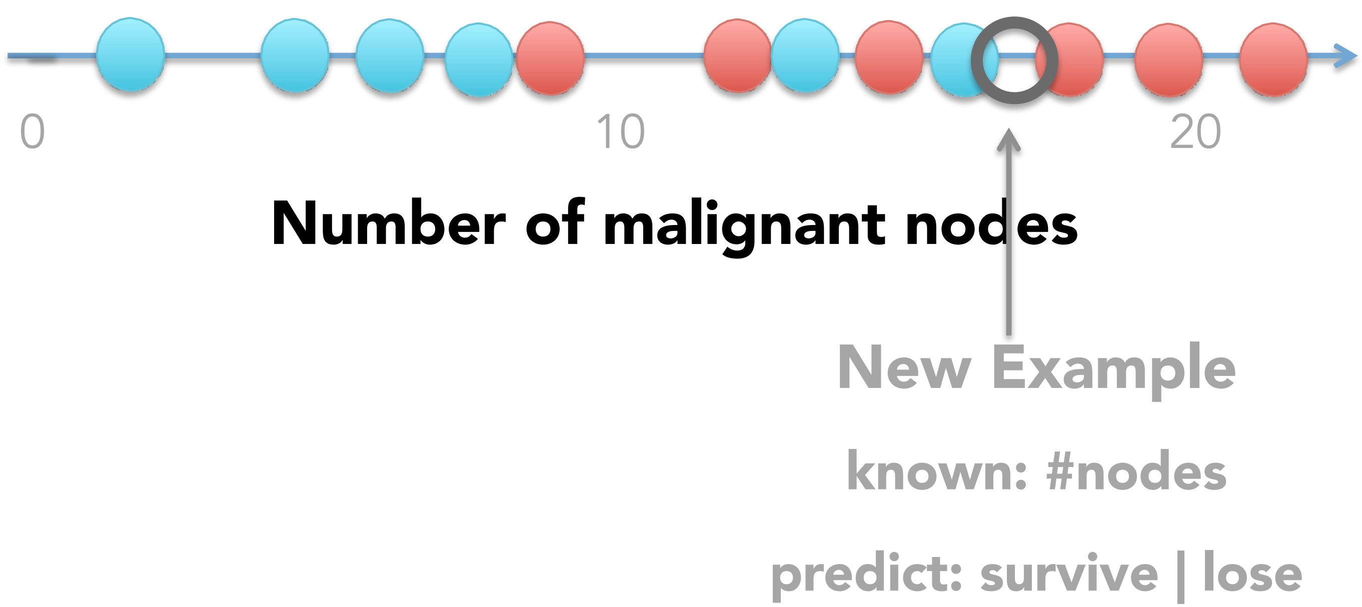
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



1 Feature. Number of malignant nodes
2 Labels. Survived / Lost

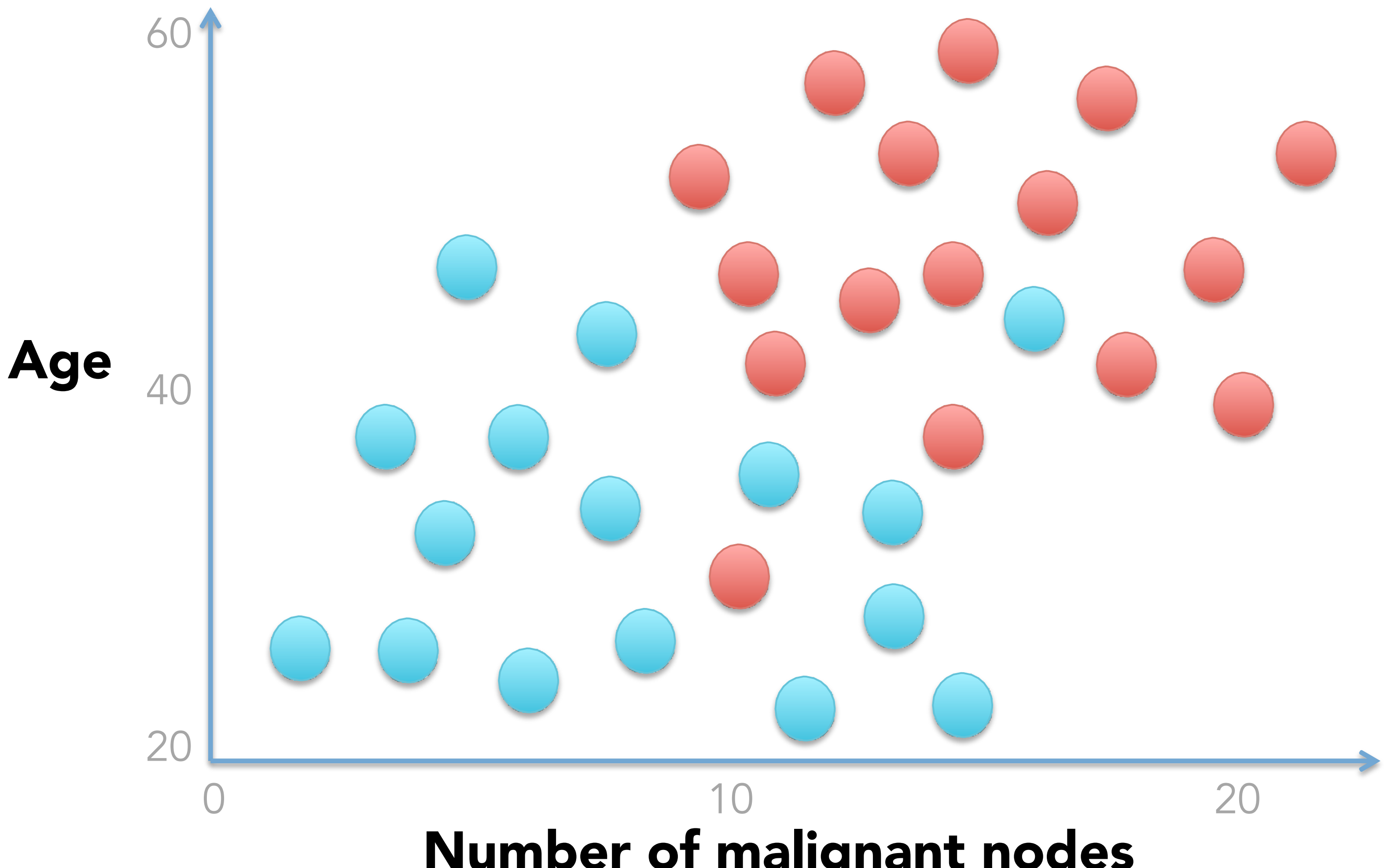


1 Feature. Number of malignant nodes
2 Labels. Survived / Lost

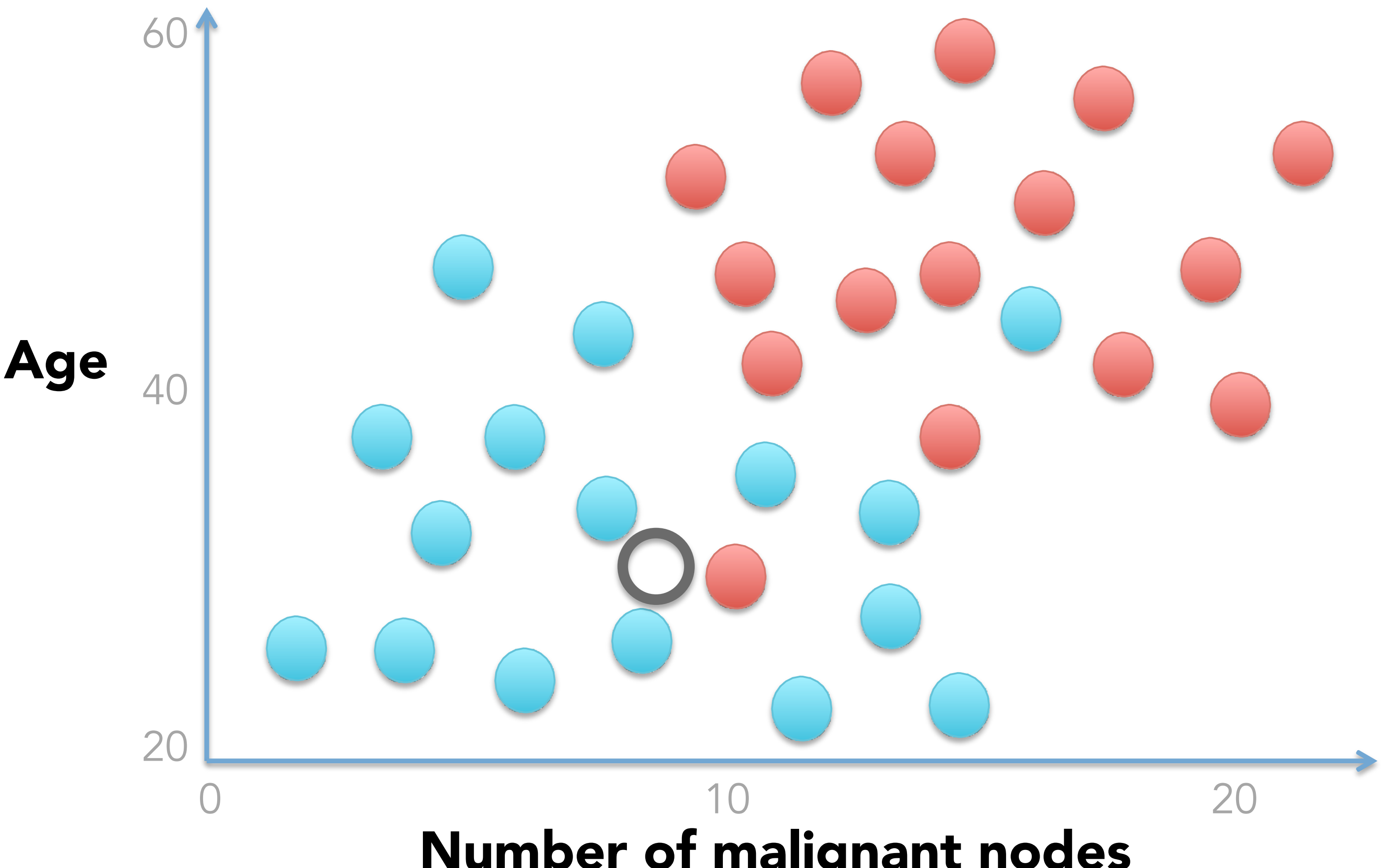


2 Features. No of malignant nodes / **Age**

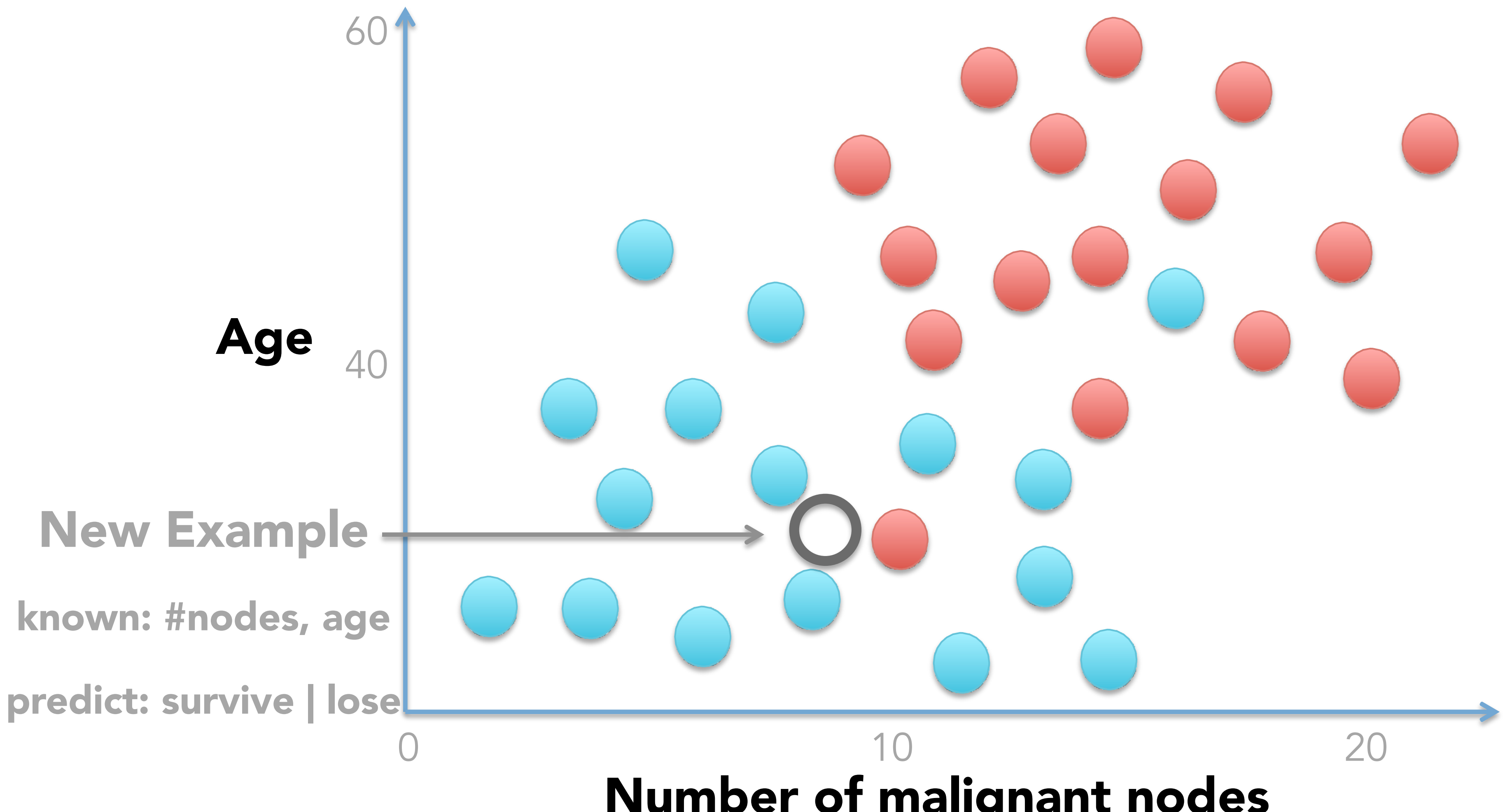
2 Labels. Survived / Lost



2 Features. No of malignant nodes / Age
2 Labels. Survived / Lost

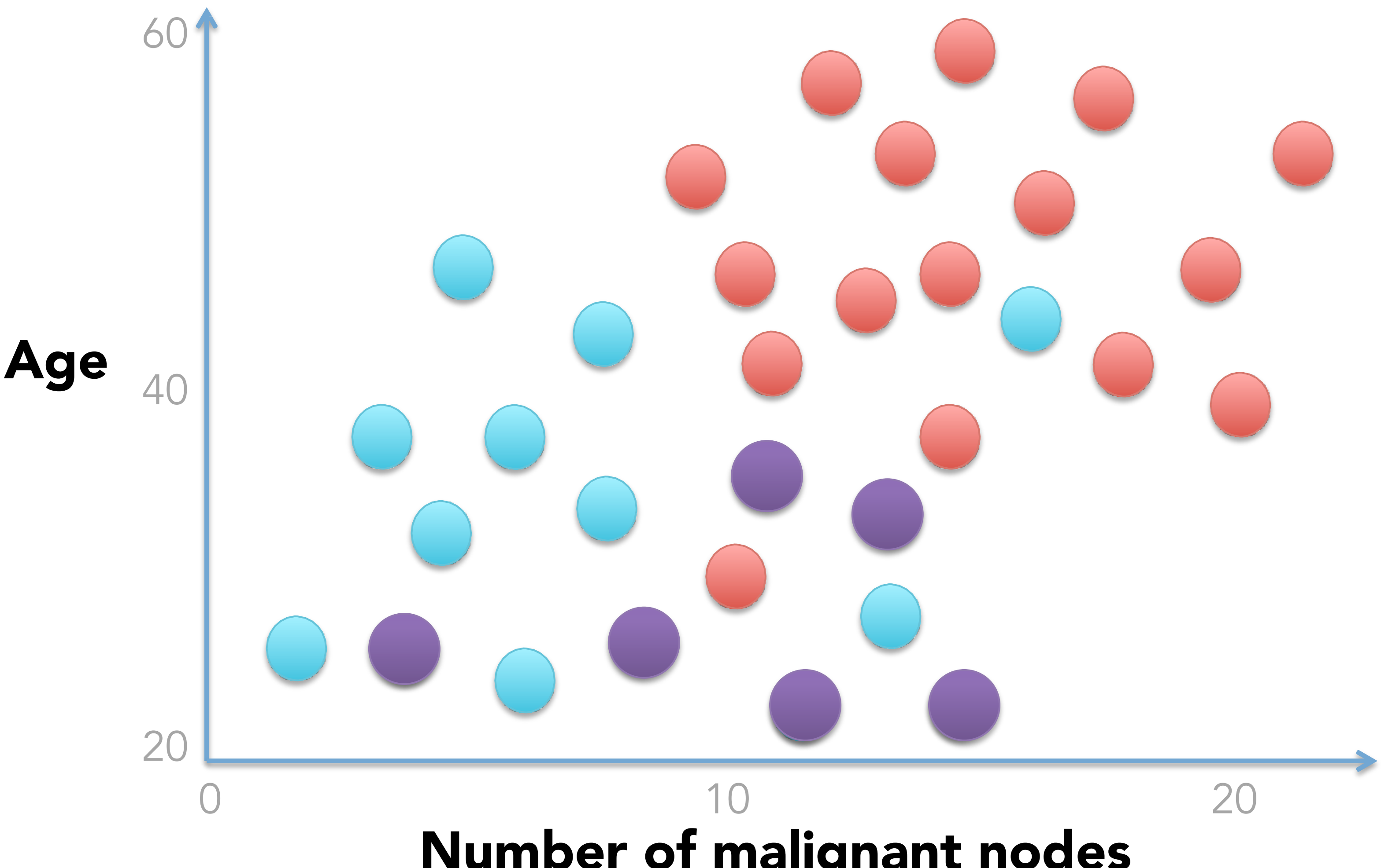


2 Features. No of malignant nodes / Age
2 Labels. Survived / Lost

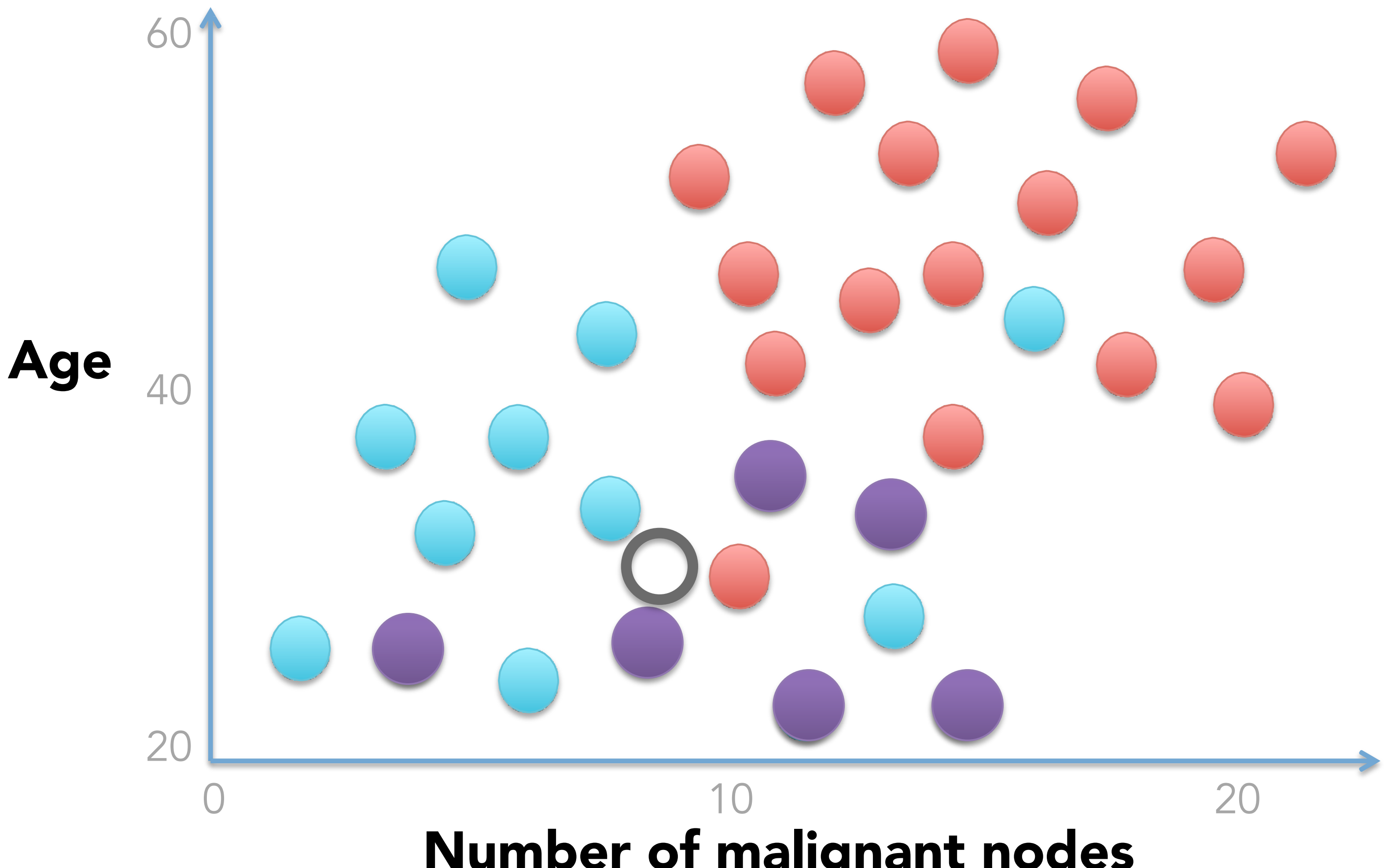


2 Features. No of malignant nodes / Age

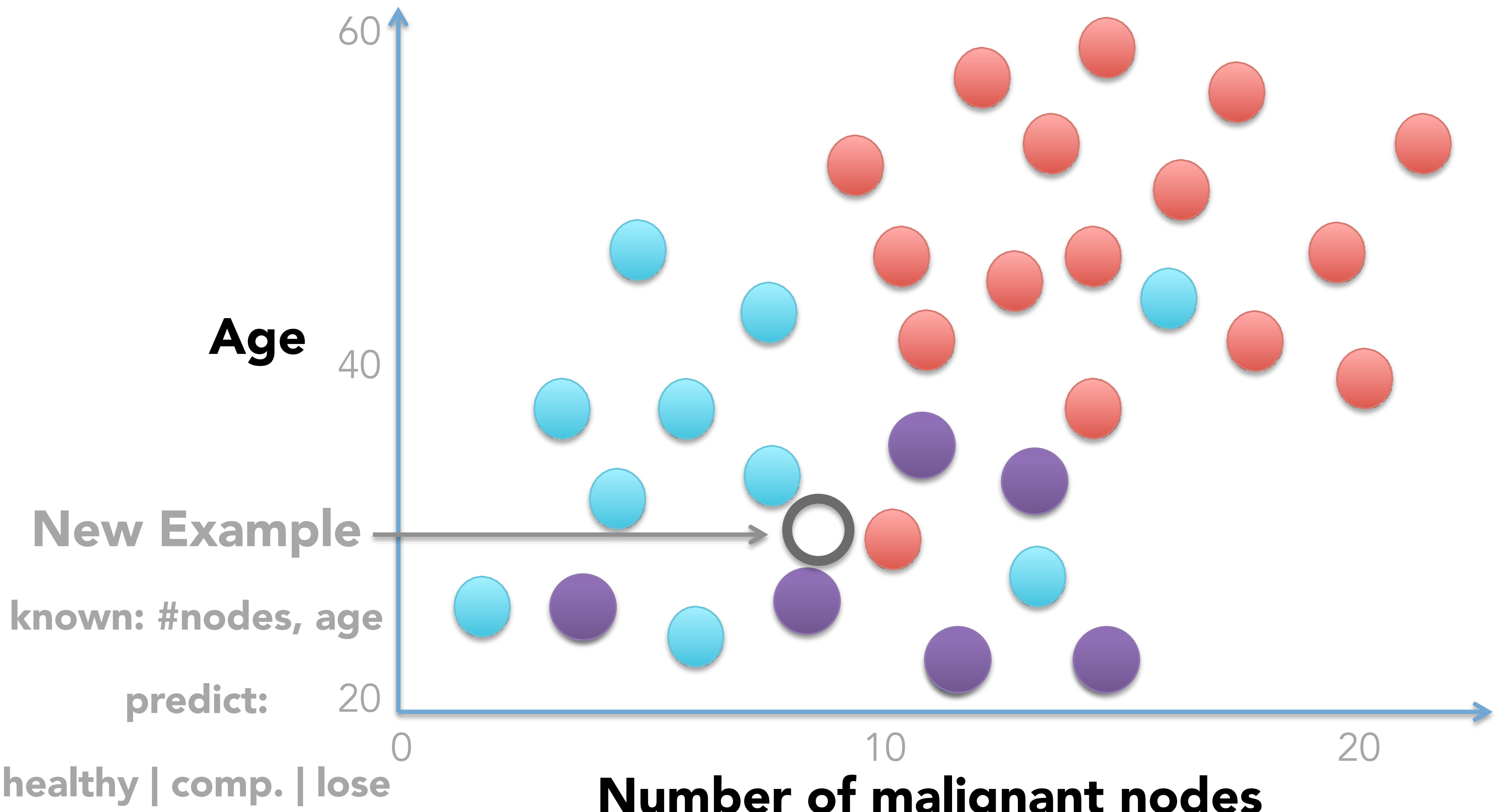
3 Labels. Healthy / Complications / Lost



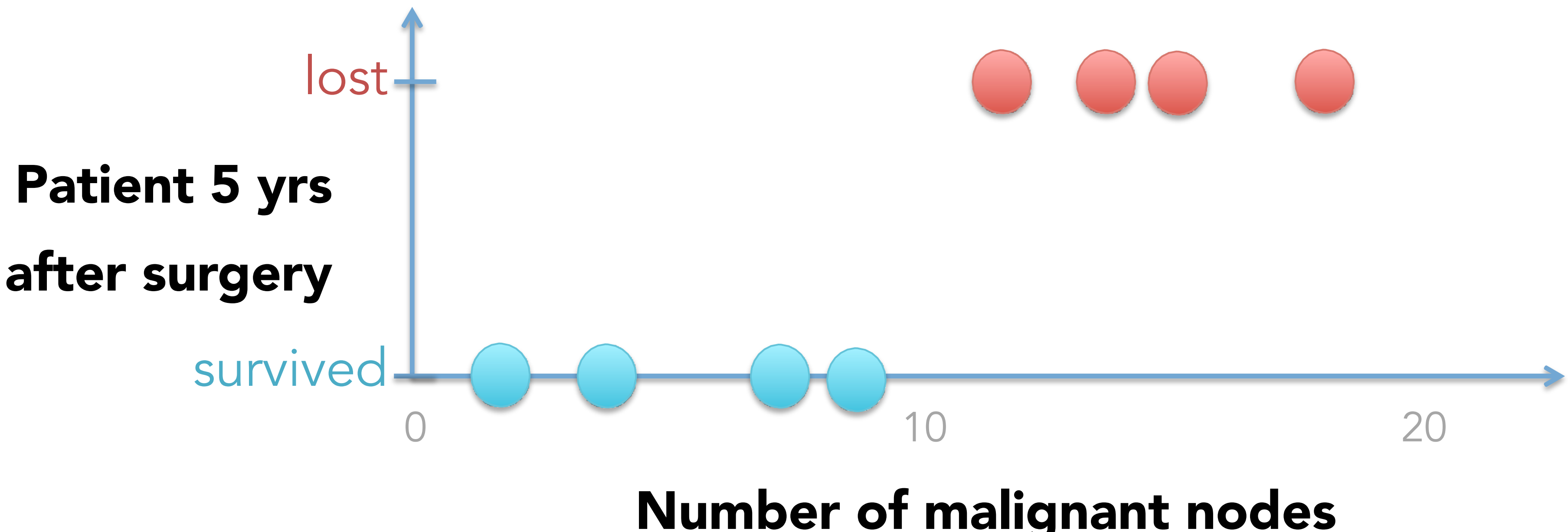
2 Features. No of malignant nodes / Age
3 Labels. Healthy / Complications / Lost



2 Features. No of malignant nodes / Age
3 Labels. Healthy / Complications / Lost

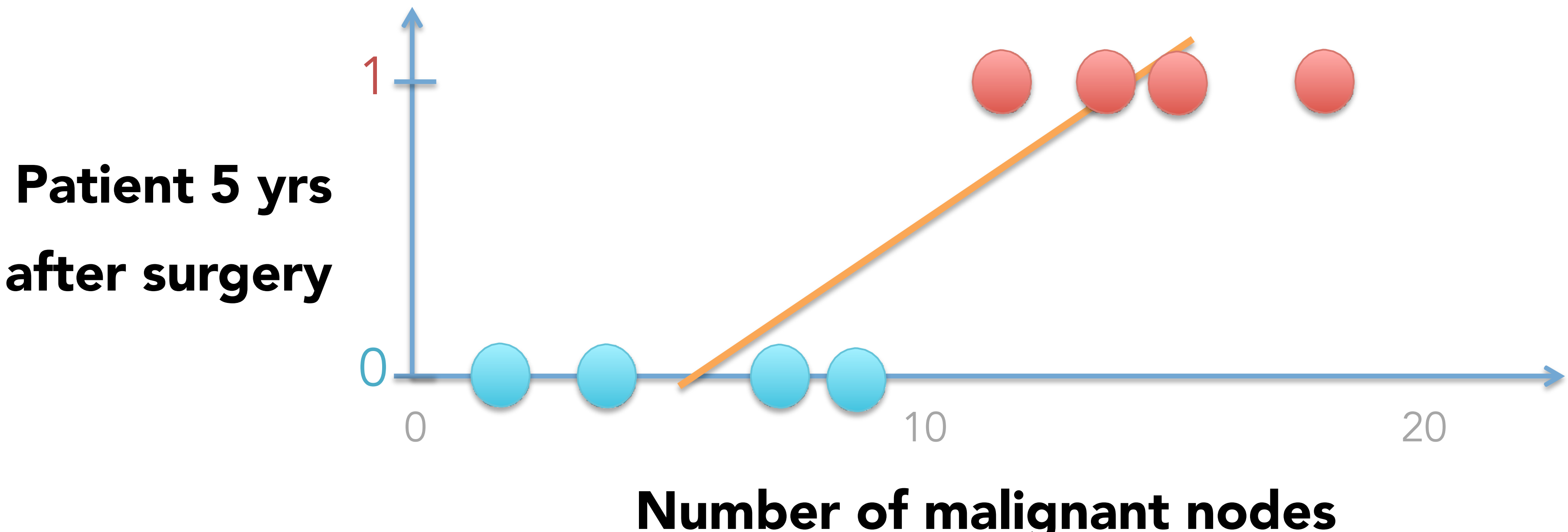


Linear Regression for classification?



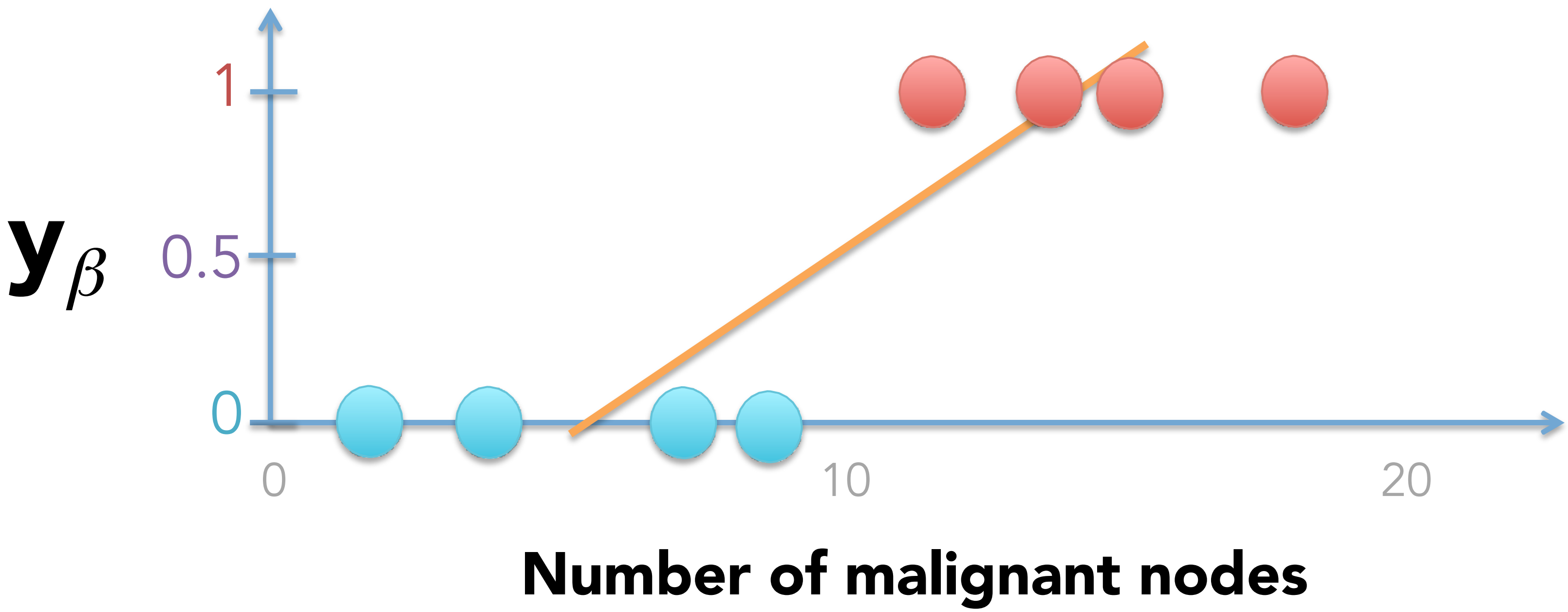
Linear Regression for classification?

$$y_\beta(x) = \beta_0 + \beta_1 x + \varepsilon$$



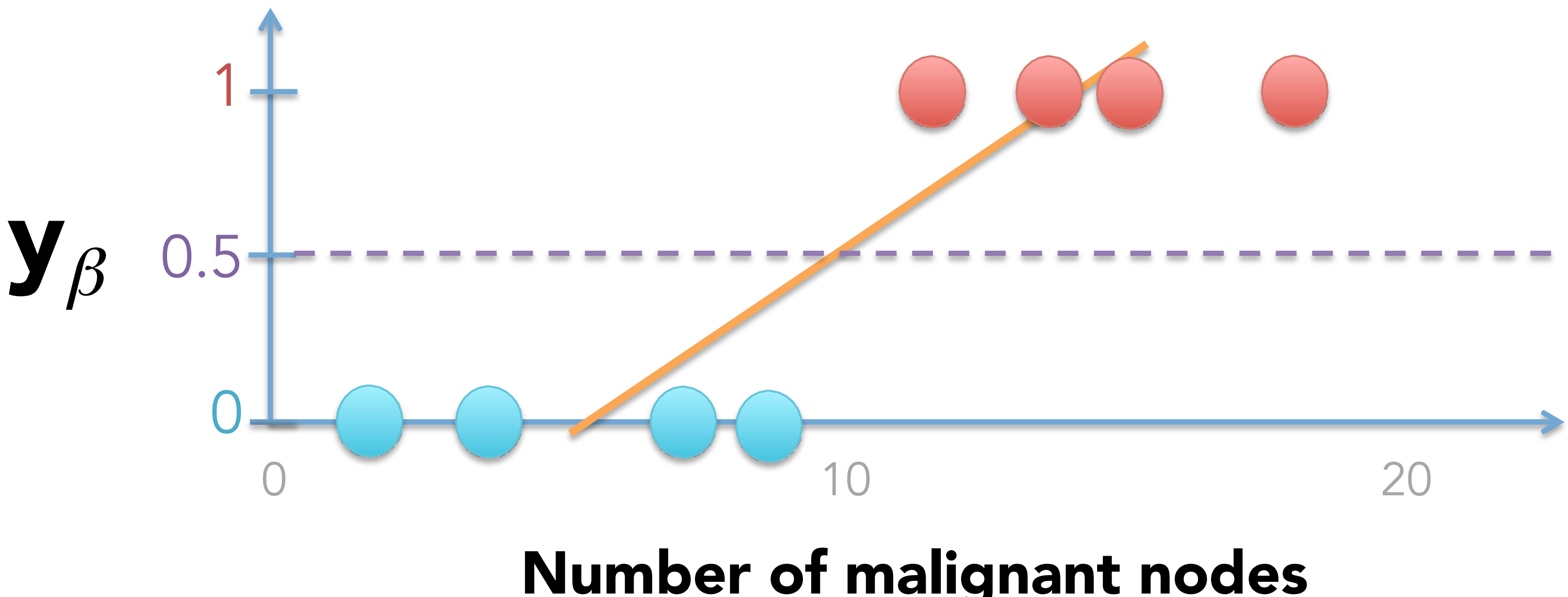
Linear Regression for classification?

$$y_\beta(\#nodes) = \beta_0 + \beta_1(\#nodes) + \varepsilon$$



Linear Regression for classification?

$$y_\beta(\#nodes) = \beta_0 + \beta_1(\#nodes) + \varepsilon$$

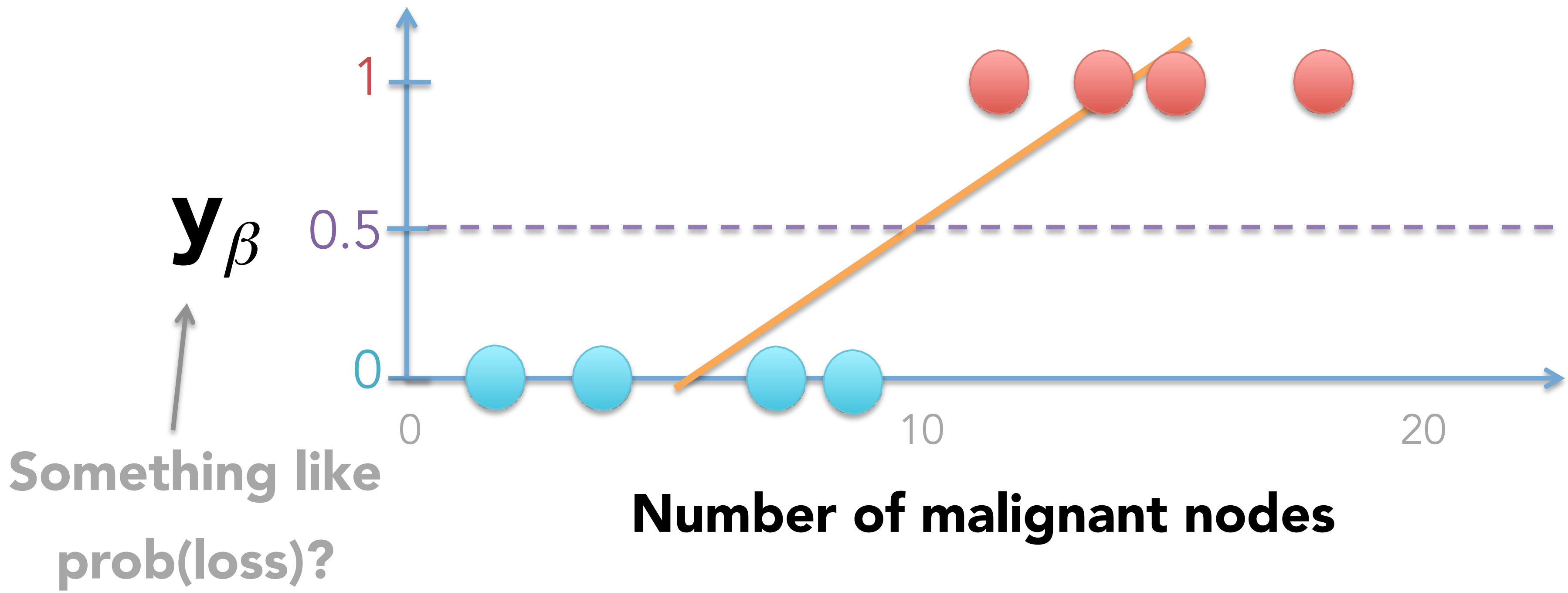


Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

Linear Regression for classification?

$$prob = \beta_0 + \beta_1(\#nodes) + \varepsilon$$

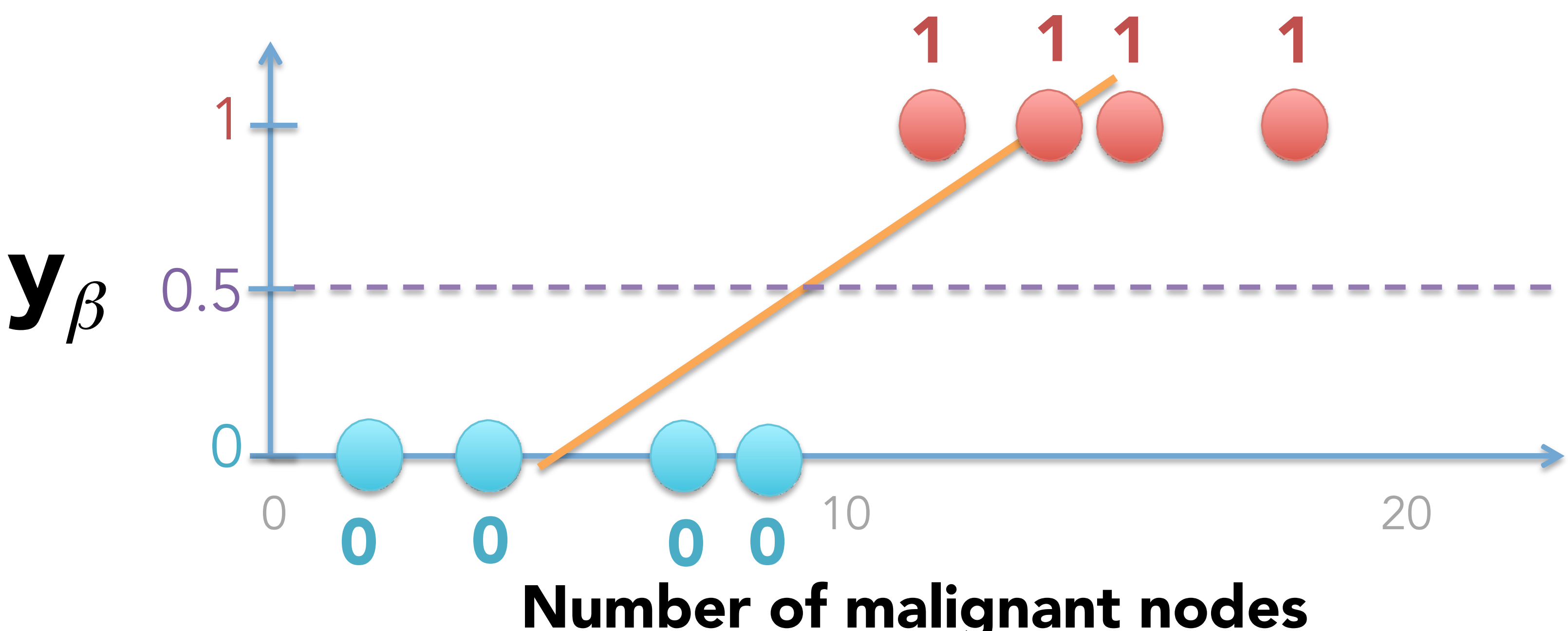


Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

Linear Regression for classification?

$$prob = \beta_0 + \beta_1(\#nodes) + \varepsilon$$

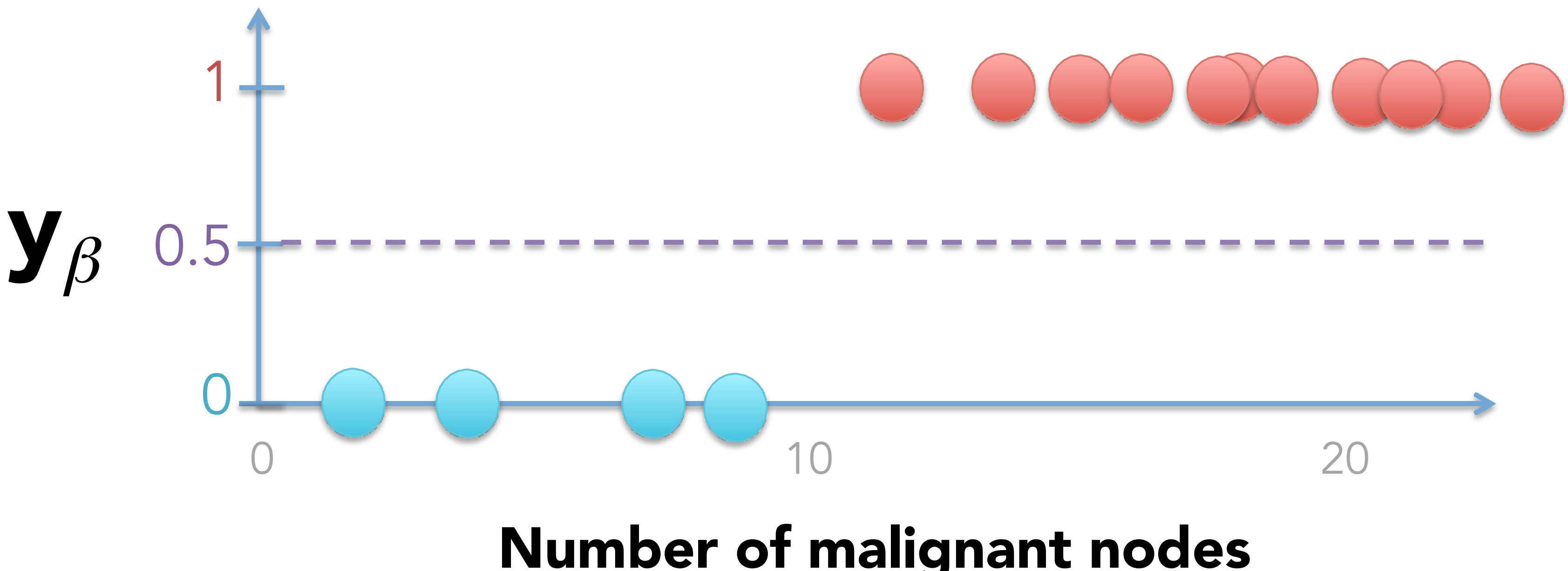


Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

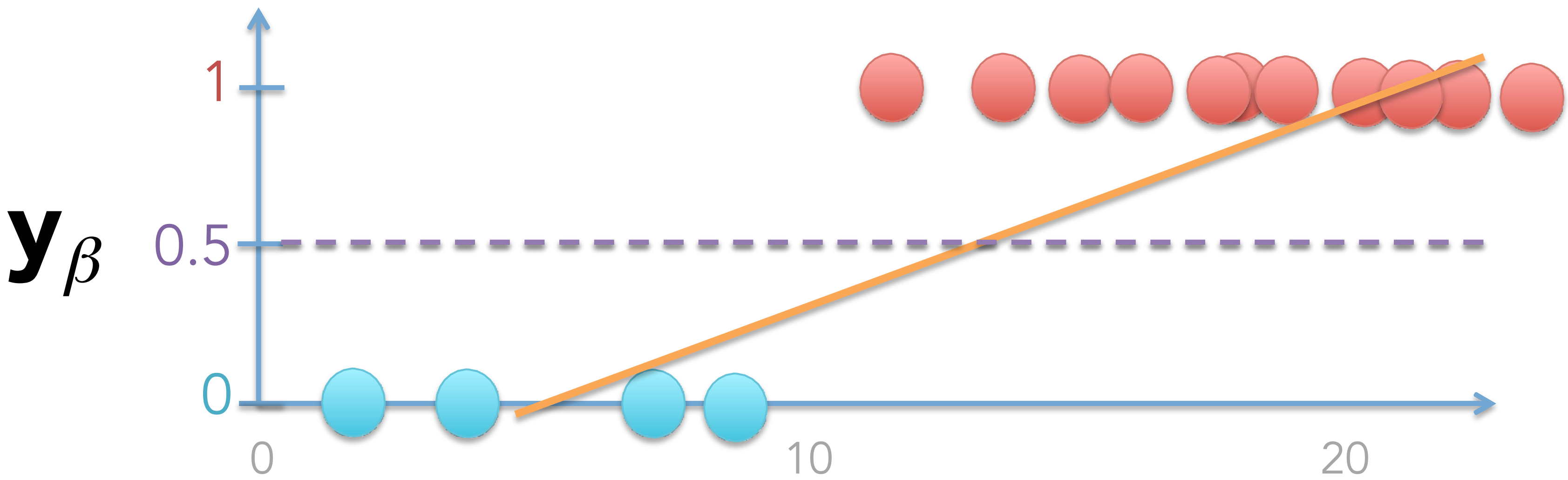
Linear Regression for classification?

More data arrives on the right end



Linear Regression for classification?

$$prob = \beta_0 + \beta_1 x + \varepsilon$$



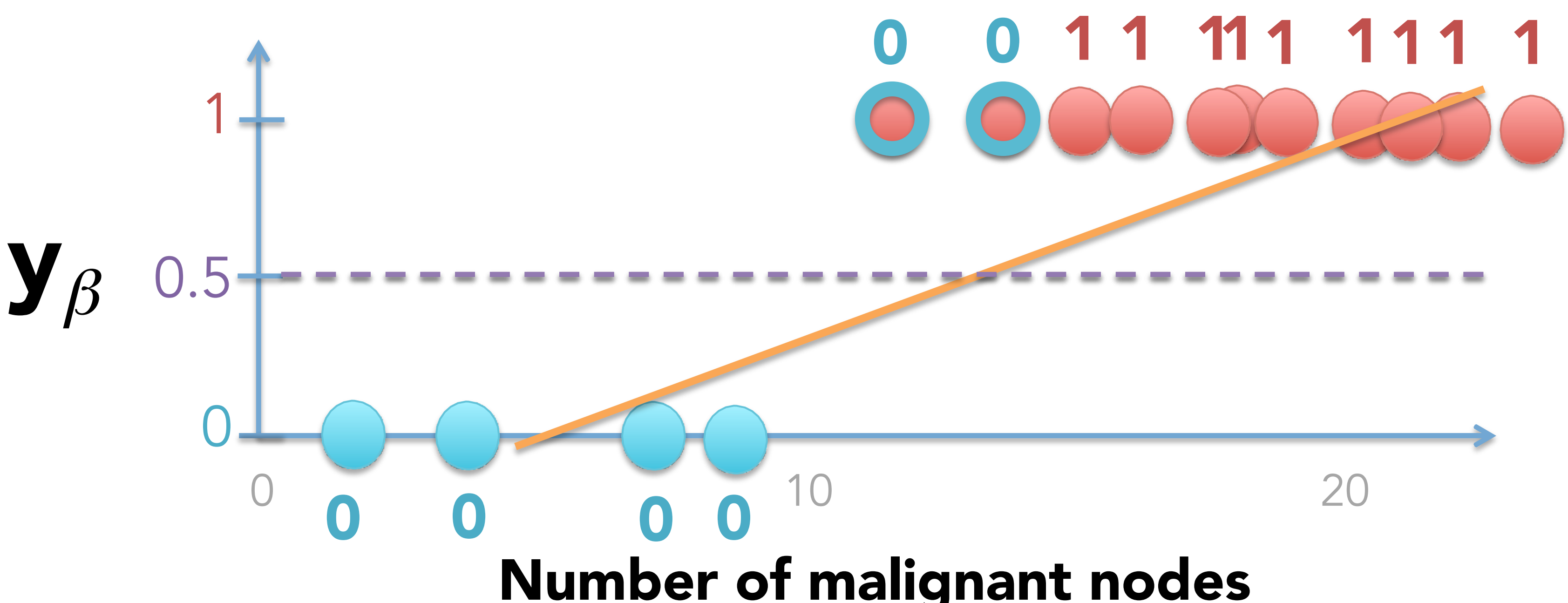
Number of malignant nodes

Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

Linear Regression for classification?

$$prob = \beta_0 + \beta_1 x + \varepsilon$$

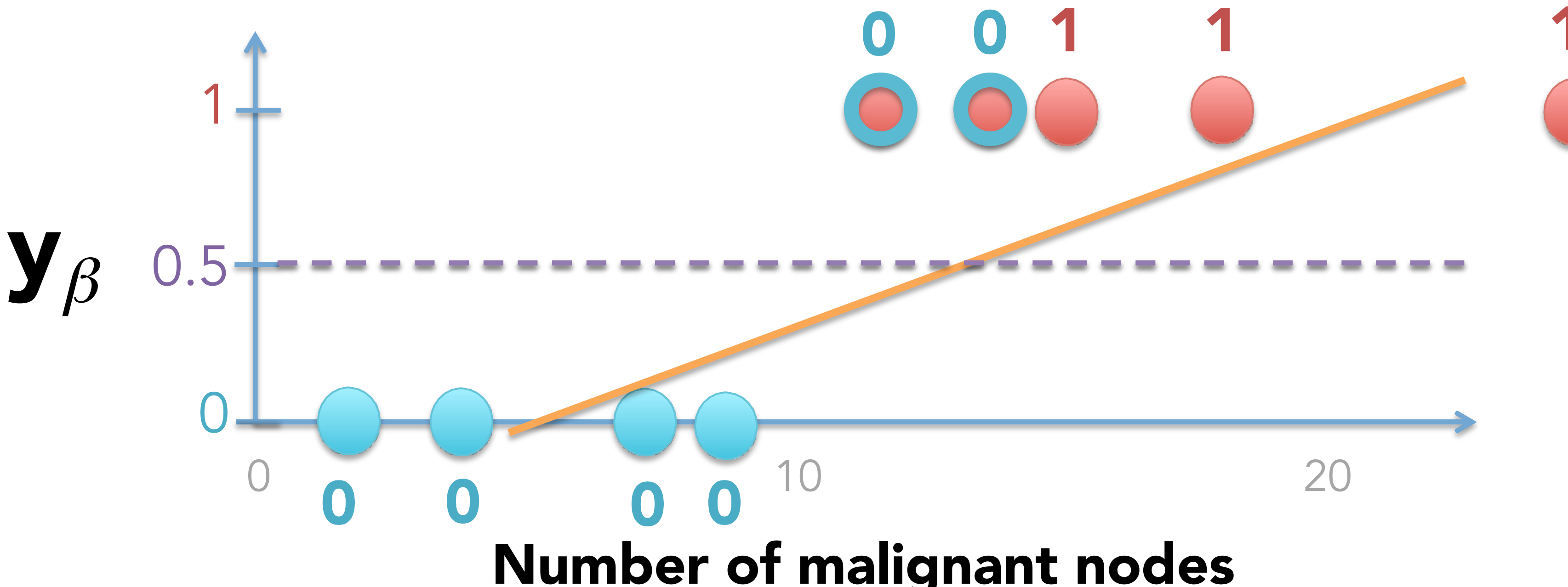


Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

Linear Regression for classification?

$$prob = \beta_0 + \beta_1 x + \varepsilon$$

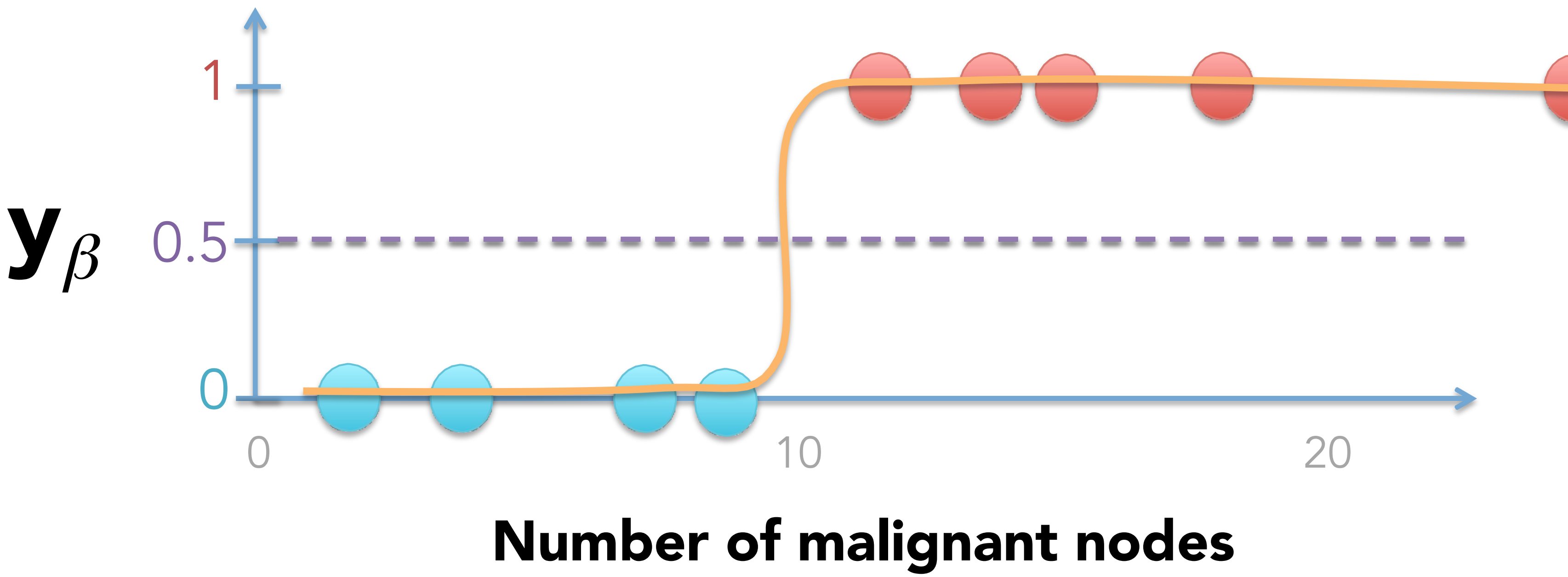


Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

Logistic Regression to the rescue

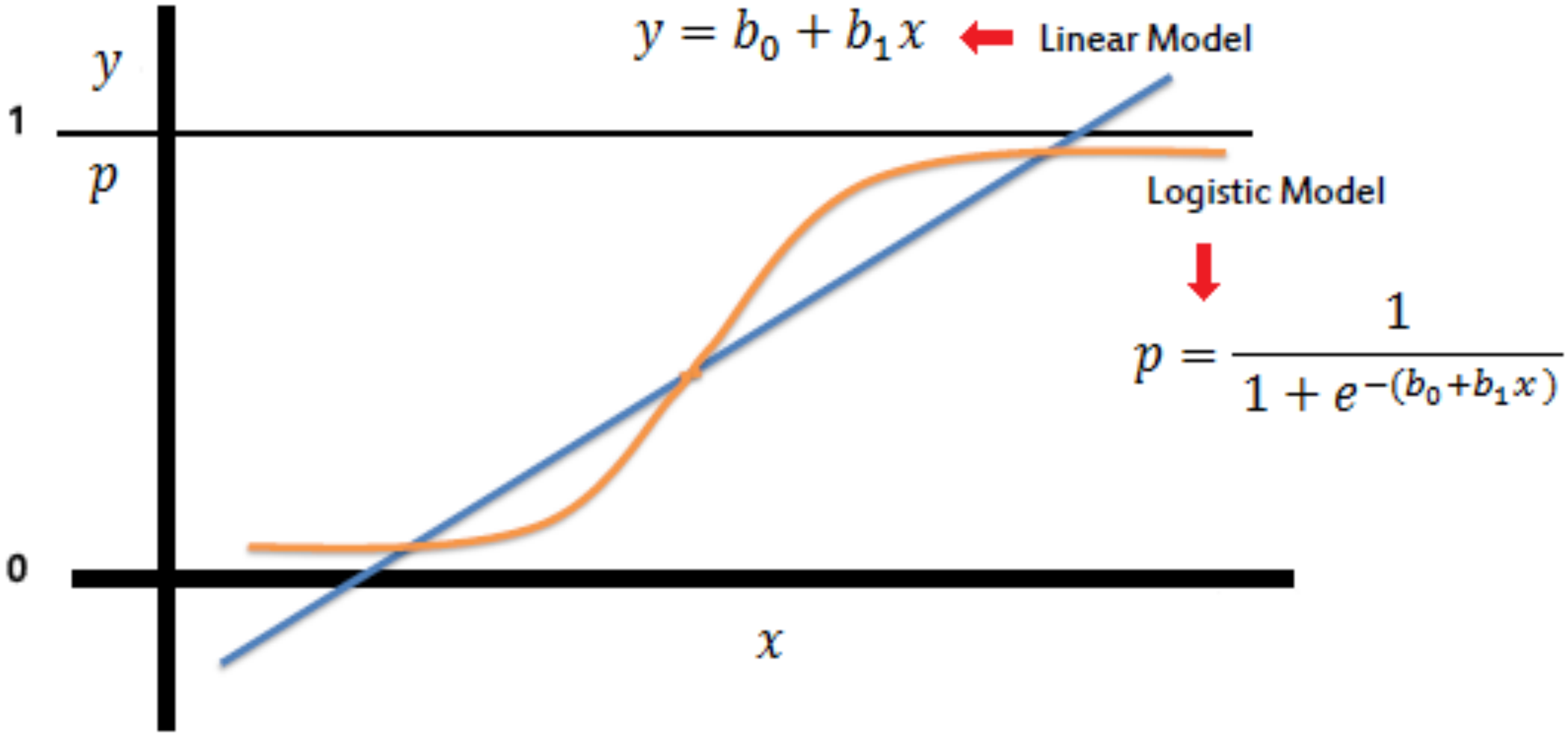
$$\textbf{logit}(prob) = \beta_0 + \beta_1 x + \varepsilon$$



Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

What is this function?



http://www.saedsayad.com/logistic_regression.htm

What is this function?

$$\mathbf{logit}(prob) = \beta_0 + \beta_1 x + \varepsilon$$

y = logit(prob)

$$prob = \frac{1}{1+e^{-(\beta_0+\beta_1 x+\varepsilon)}}$$

prob = logistic(y)

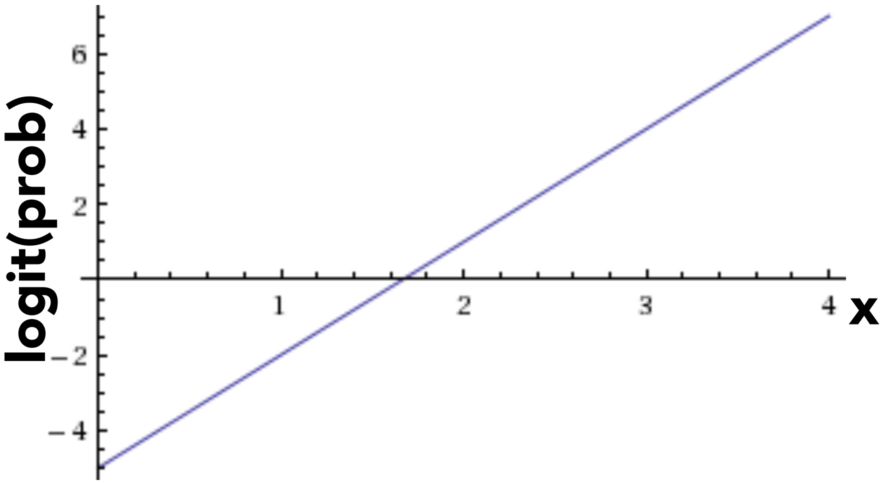
What is this function?

$$\text{logit}(prob) = \beta_0 + \beta_1 x + \varepsilon$$

y = logit(prob)

$$prob = \frac{1}{1+e^{-(\beta_0+\beta_1 x+\varepsilon)}}$$

prob = logistic(y)



x

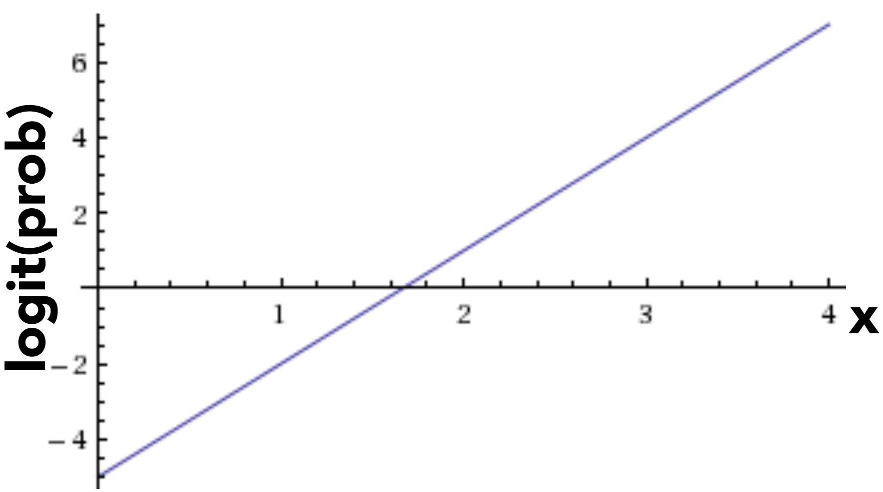
What is this function?

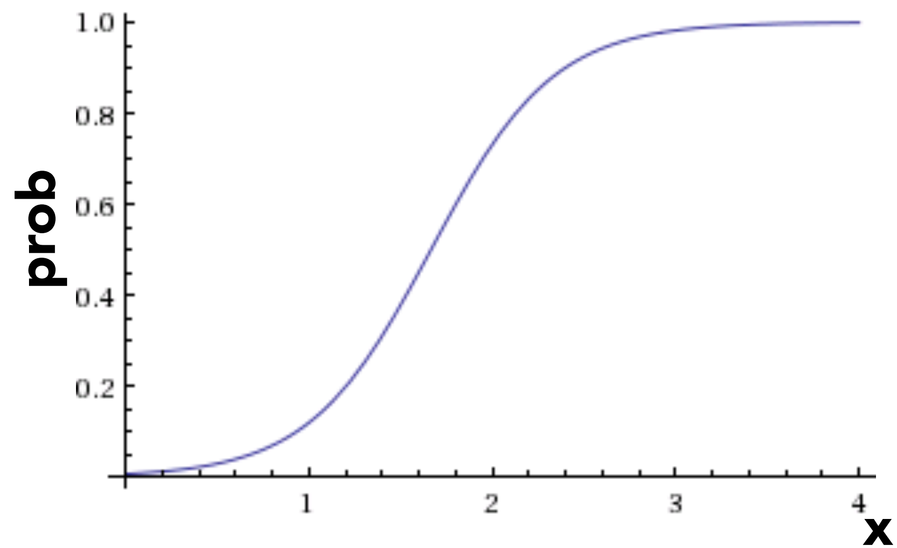
$$\text{logit}(prob) = \beta_0 + \beta_1 x + \varepsilon$$

y = logit(prob)

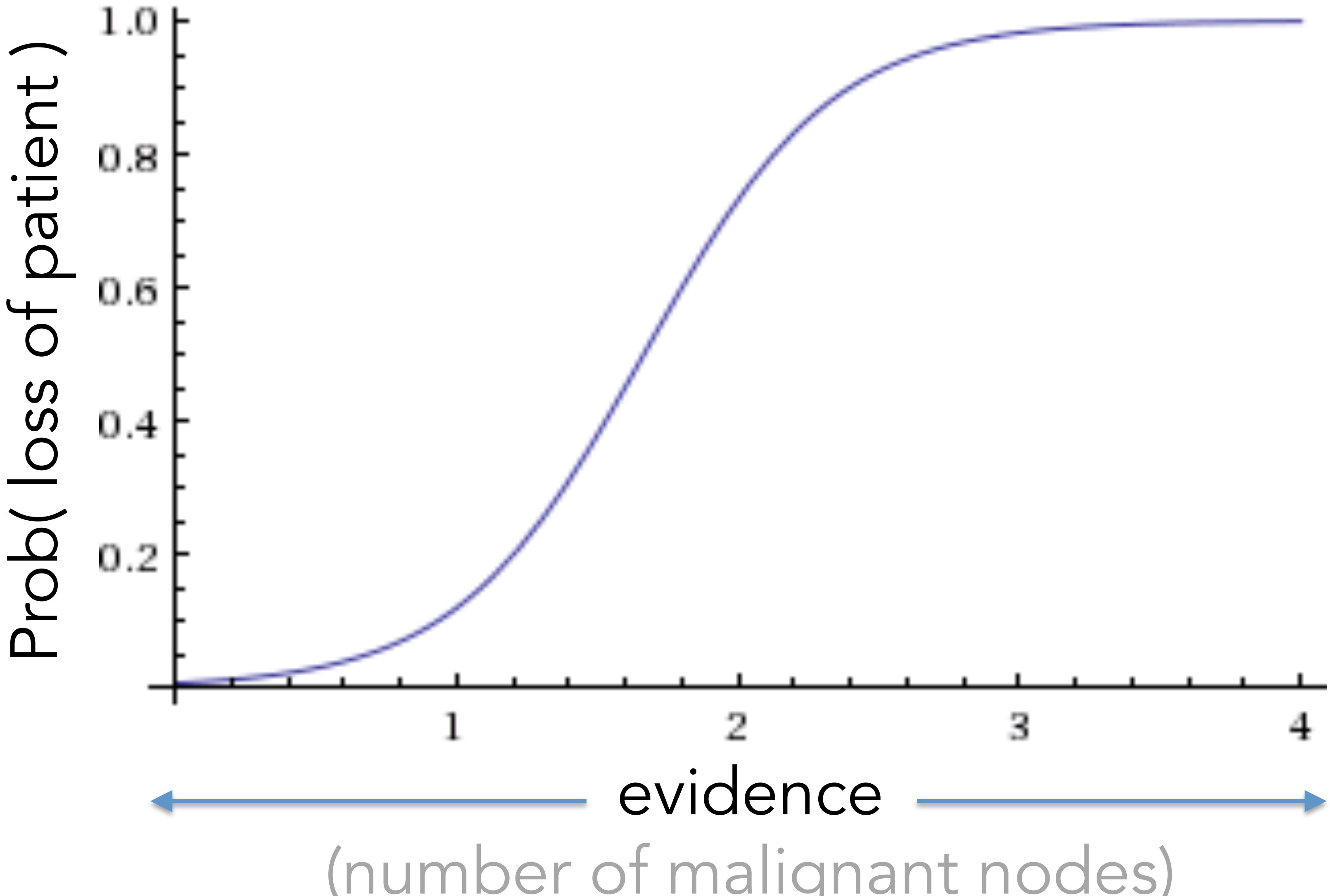
$$prob = \frac{1}{1+e^{-(\beta_0+\beta_1 x+\varepsilon)}}$$

prob = logistic(y)

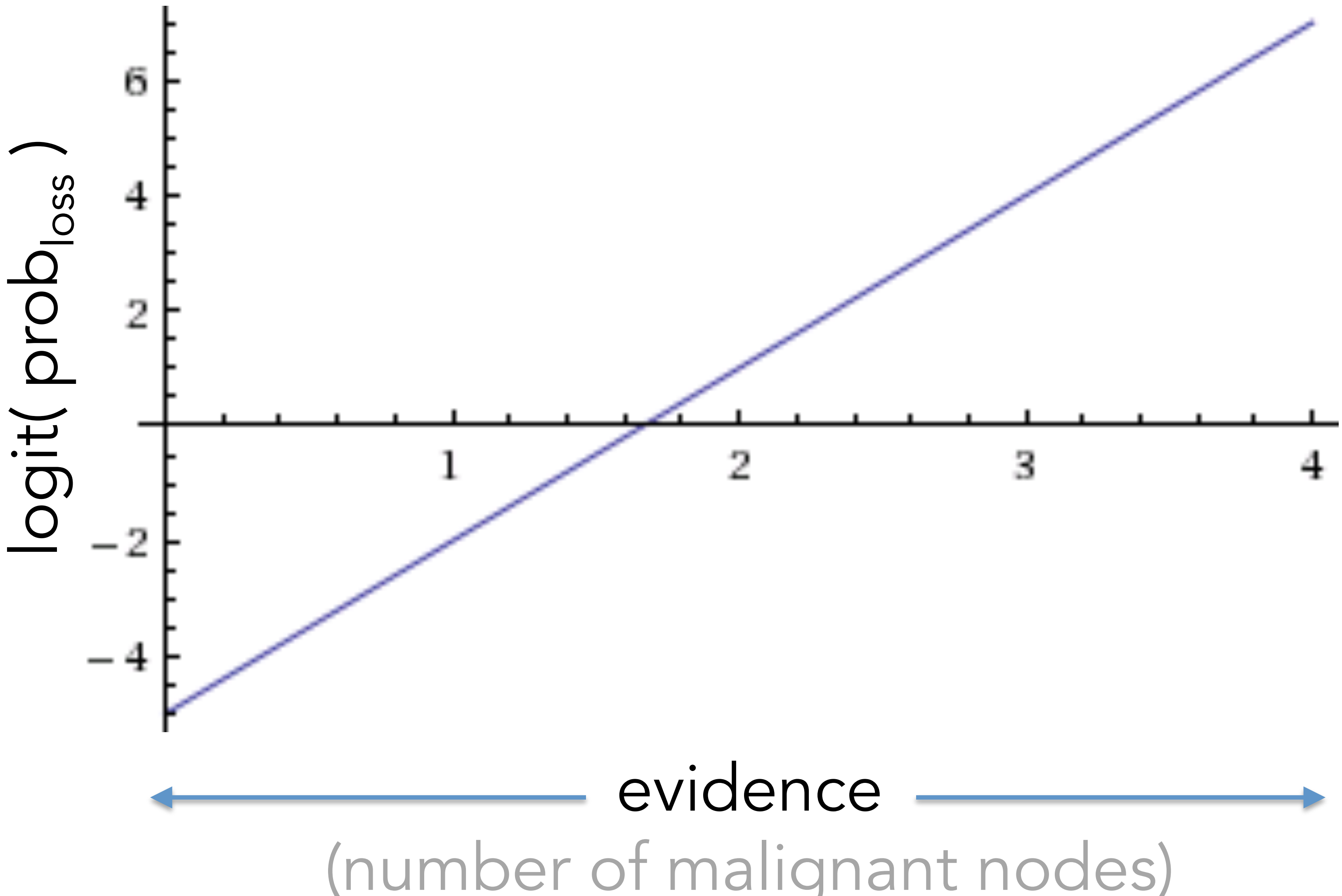




What is this function?

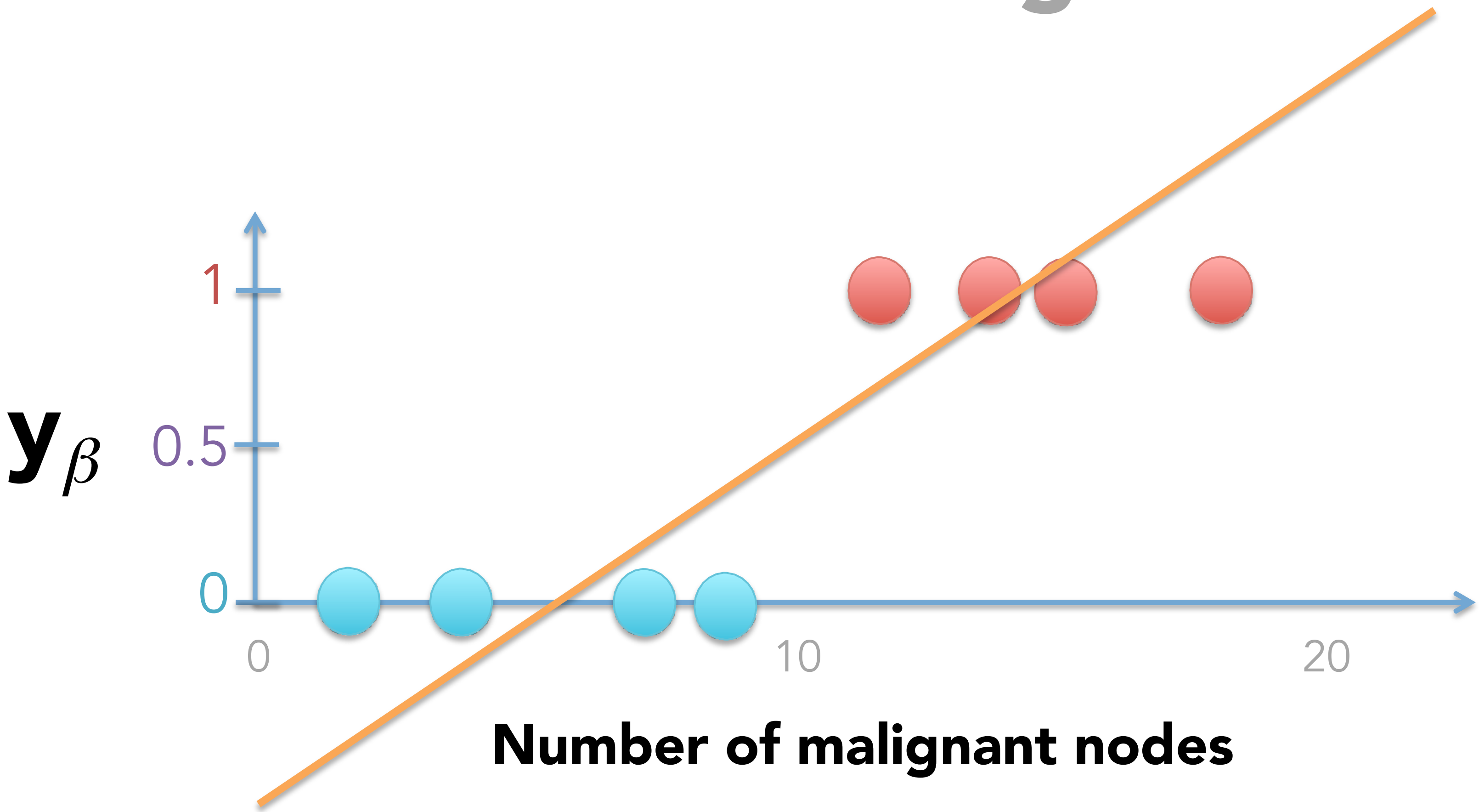


What is this function?



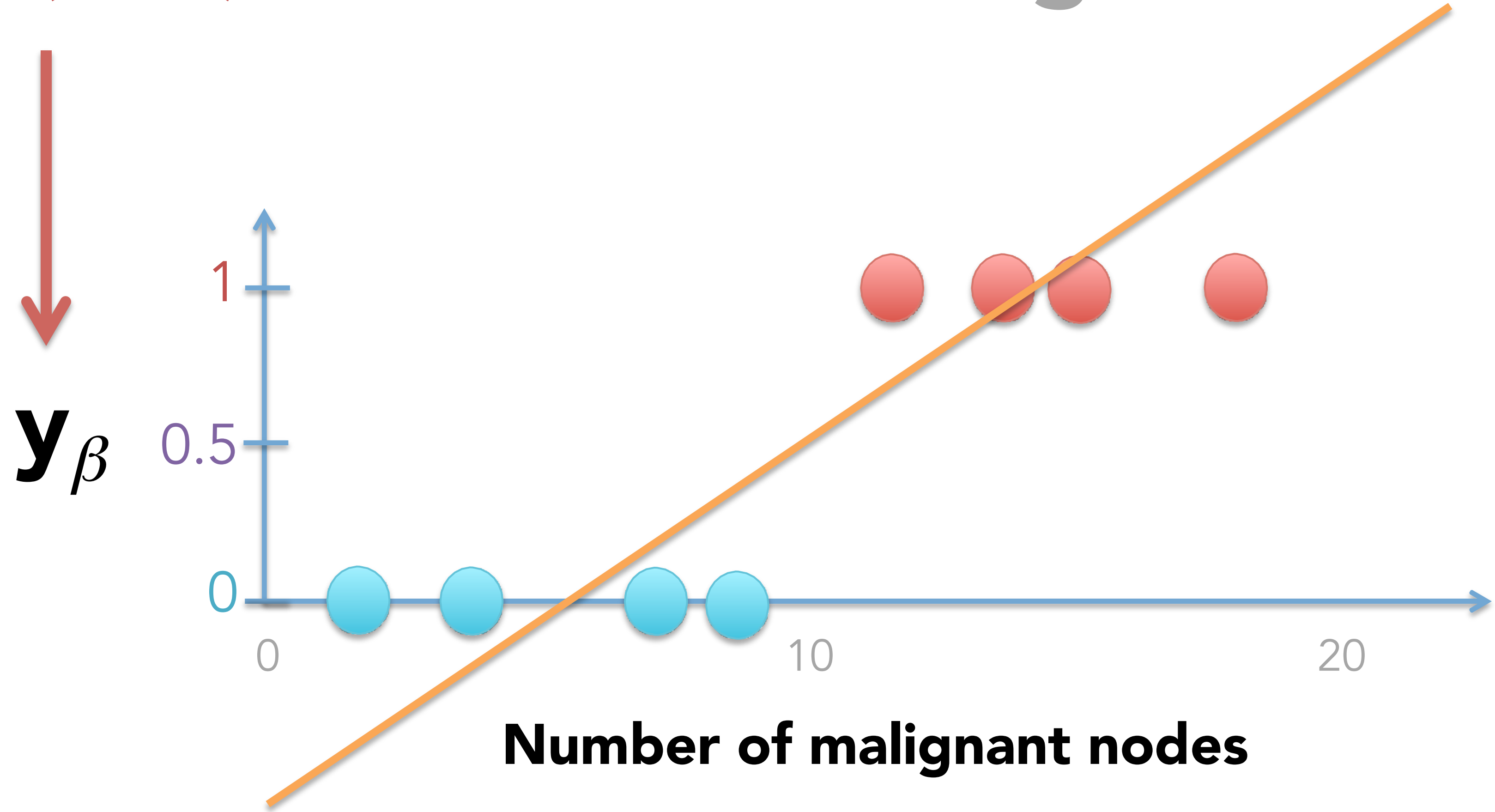
Ok, but what is logit(prob)?

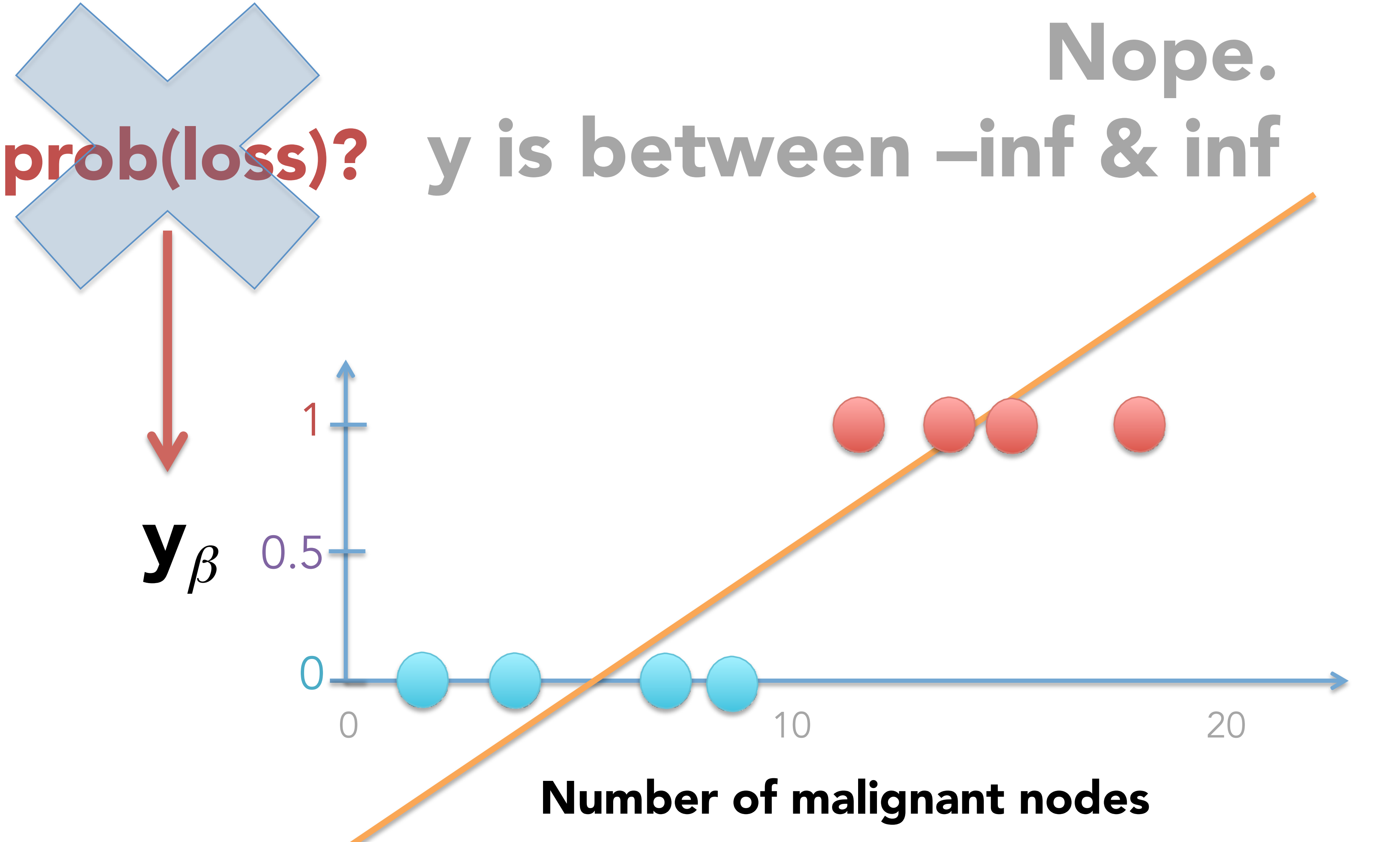
Back to trying linear regression



Back to trying linear regression

prob(loss)?




Nope.
y is between –inf & inf
prob(loss)?
$\mathbf{y}_\beta$
1
0.5
0
0
10
20
Number of malignant nodes

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.8$$

$$P(survival) = 0.2$$

Probability

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.8$$

$$P(survival) = 0.2$$

Probability

$$\frac{P(loss)}{P(survival)} = 4$$

Odds

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.05$$

$$P(survival) = 0.95$$

$$\frac{P(loss)}{P(survival)} = 0.053$$

Probability

Odds

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.5$$

$$P(survival) = 0.5$$

Probability

$$\frac{P(loss)}{P(survival)} = 1$$

Odds

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.5$$

$$P(survival) = 0.5$$

Probability

$$\frac{P(loss)}{P(survival)} = 1$$

Odds

between 0 and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.5$$

$$P(survival) = 0.5$$

Probability

$$\log\left(\frac{P(loss)}{P(survival)}\right) = 0$$

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.05$$

$$P(survival) = 0.95$$

$$\log\left(\frac{P(loss)}{P(survival)}\right) = -2.94$$

Probability

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.8$$

$$P(survival) = 0.2$$

Probability

$$\log\left(\frac{P(loss)}{P(survival)}\right) = 1.39$$

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.999$$

$$P(survival) = 0.001$$

Probability

$$\log\left(\frac{P(loss)}{P(survival)}\right) = 6.9$$

Log Odds

between -inf and inf

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.999$$

$$1 - P(loss) = 0.001$$

$$\log\left(\frac{P(loss)}{1 - P(loss)}\right) = 6.9$$

Probability

Log Odds

logit function

What metric would express the chances of loss/survival, but not constrained to [0, 1] ?

$$P(loss) = 0.999$$

$$1 - P(loss) = 0.001$$

Probability

$$\log\left(\frac{P(loss)}{1 - P(loss)}\right) = 6.9$$

Log Odds

logit function

$$\frac{1}{1 - e^{\log\left(\frac{P(loss)}{1-P(loss)}\right)}} = P(loss)$$

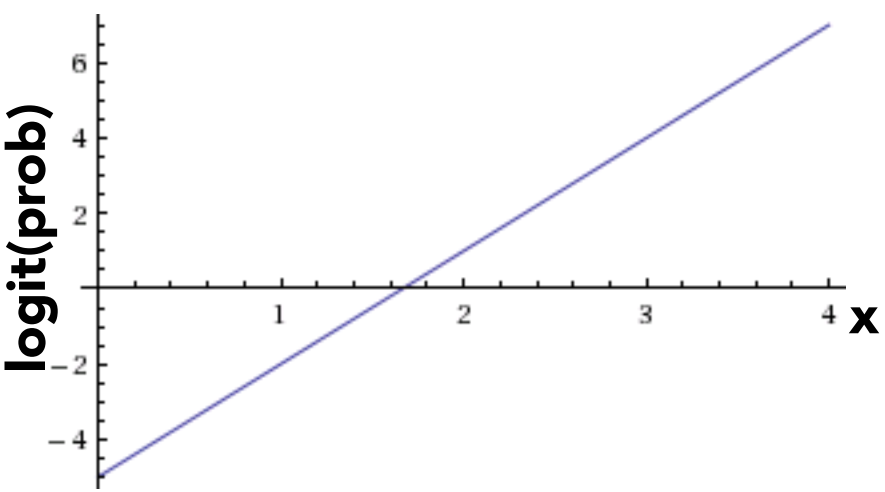
Logistic Function

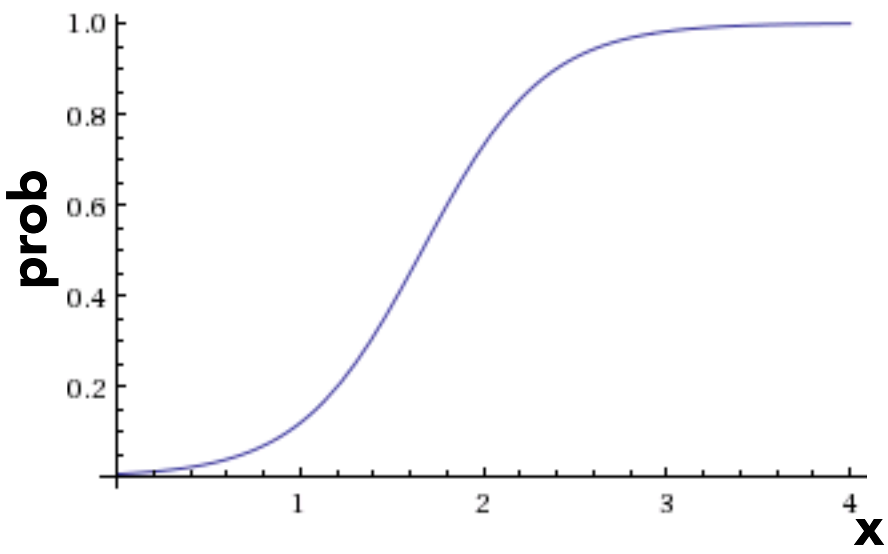
Log Odds → Prob

What is this function?

logit(prob **)** = log odds

logistic(log odds **)** = prob





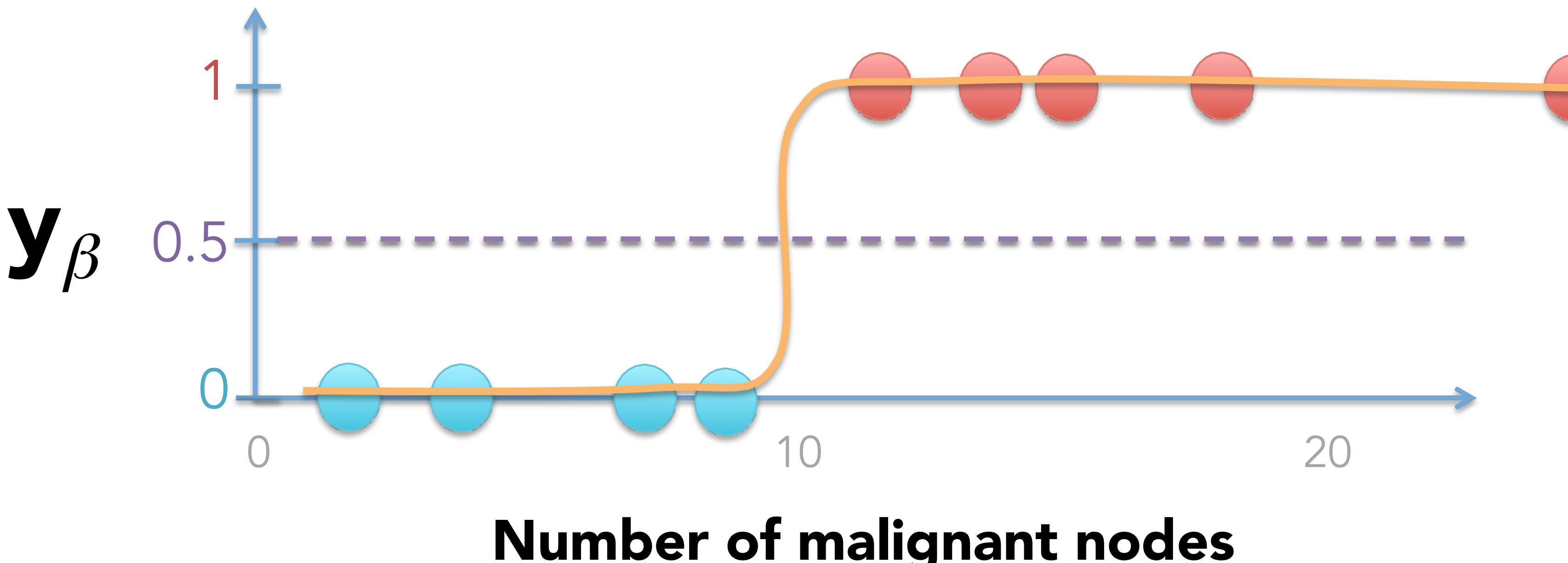
Coefficients work the same way

$$\textbf{logit(}\text{prob}\textbf{)} = \text{log odds} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \varepsilon$$

increase in log odds
per 1 unit of x_1

increase in log odds
per 1 unit of x_2

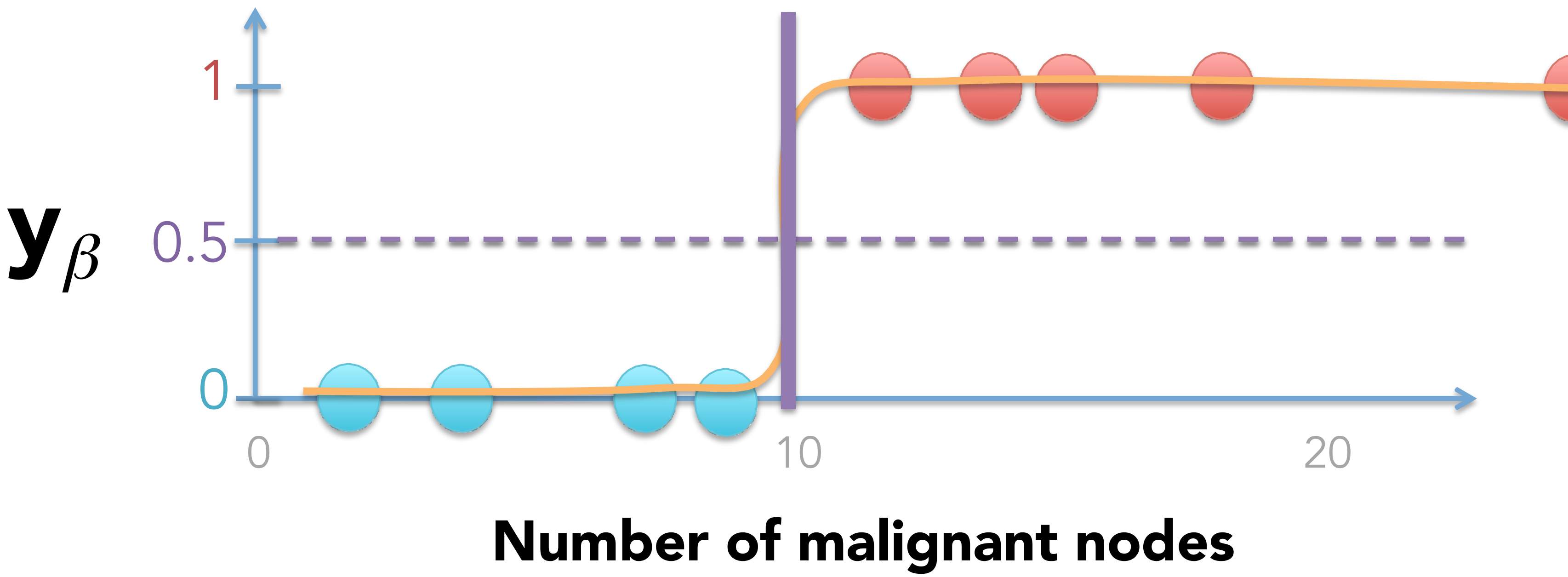
The “Decision Boundary”



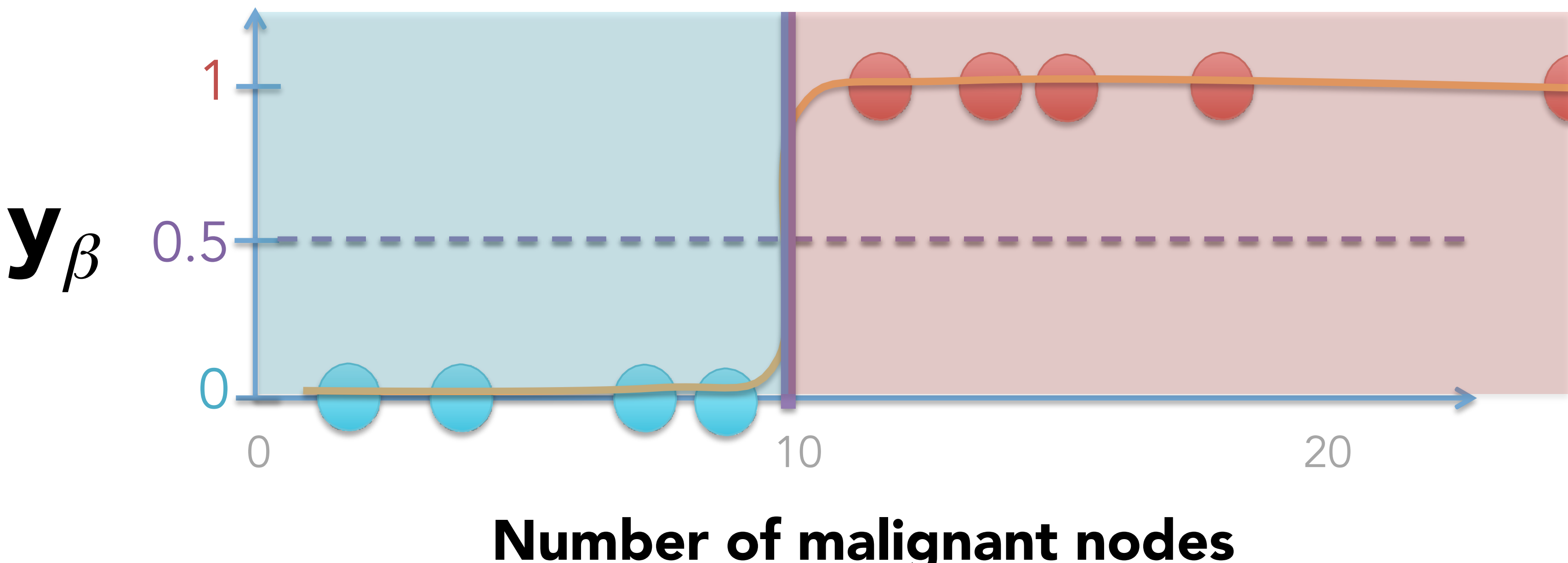
Predict 1 (lost) if $y_\beta > 0.5$

Predict 0 (survived) if $y_\beta < 0.5$

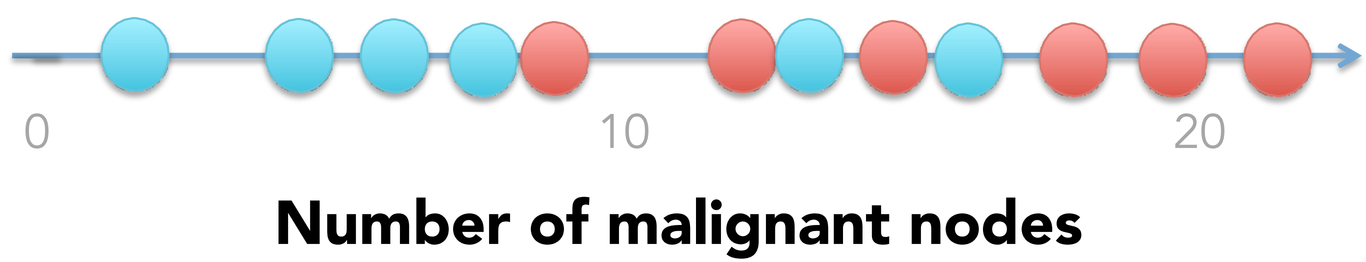
The "Decision Boundary"



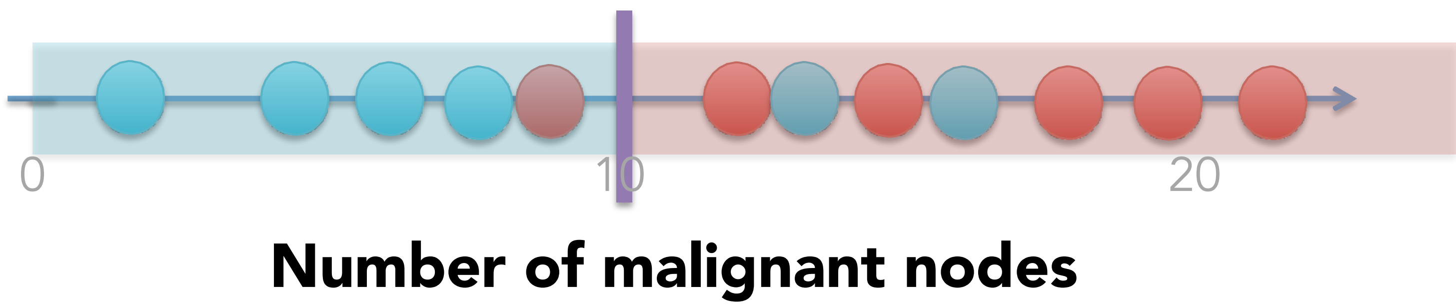
The "Decision Boundary"



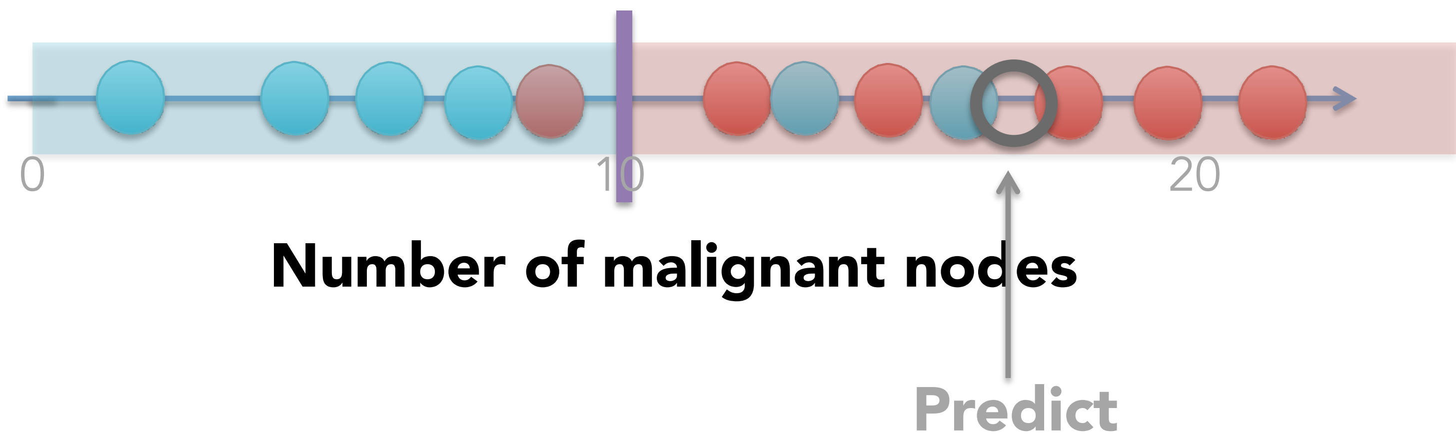
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



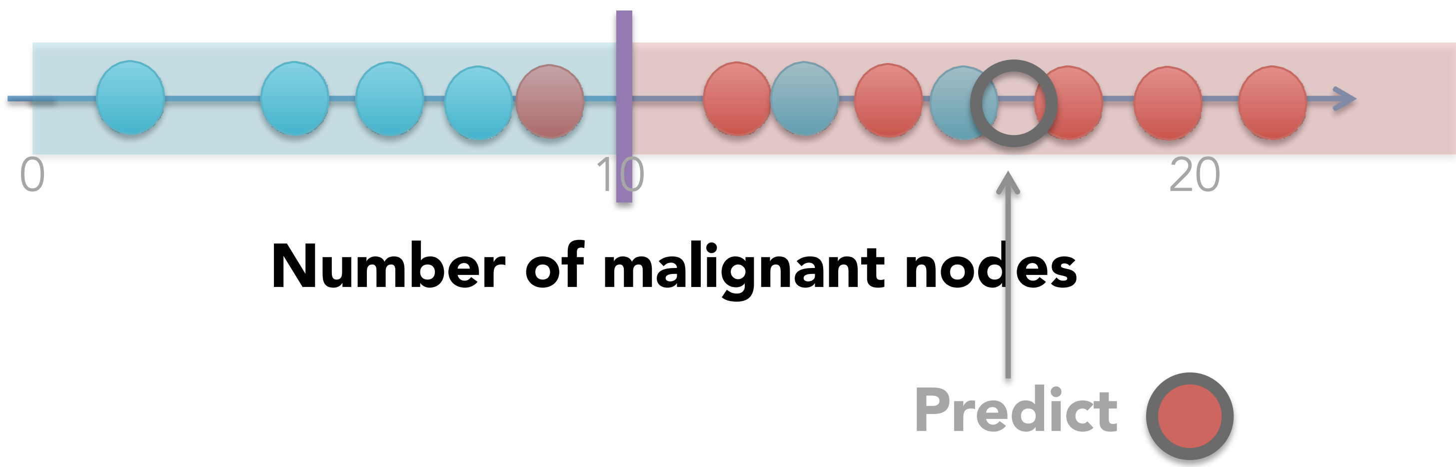
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



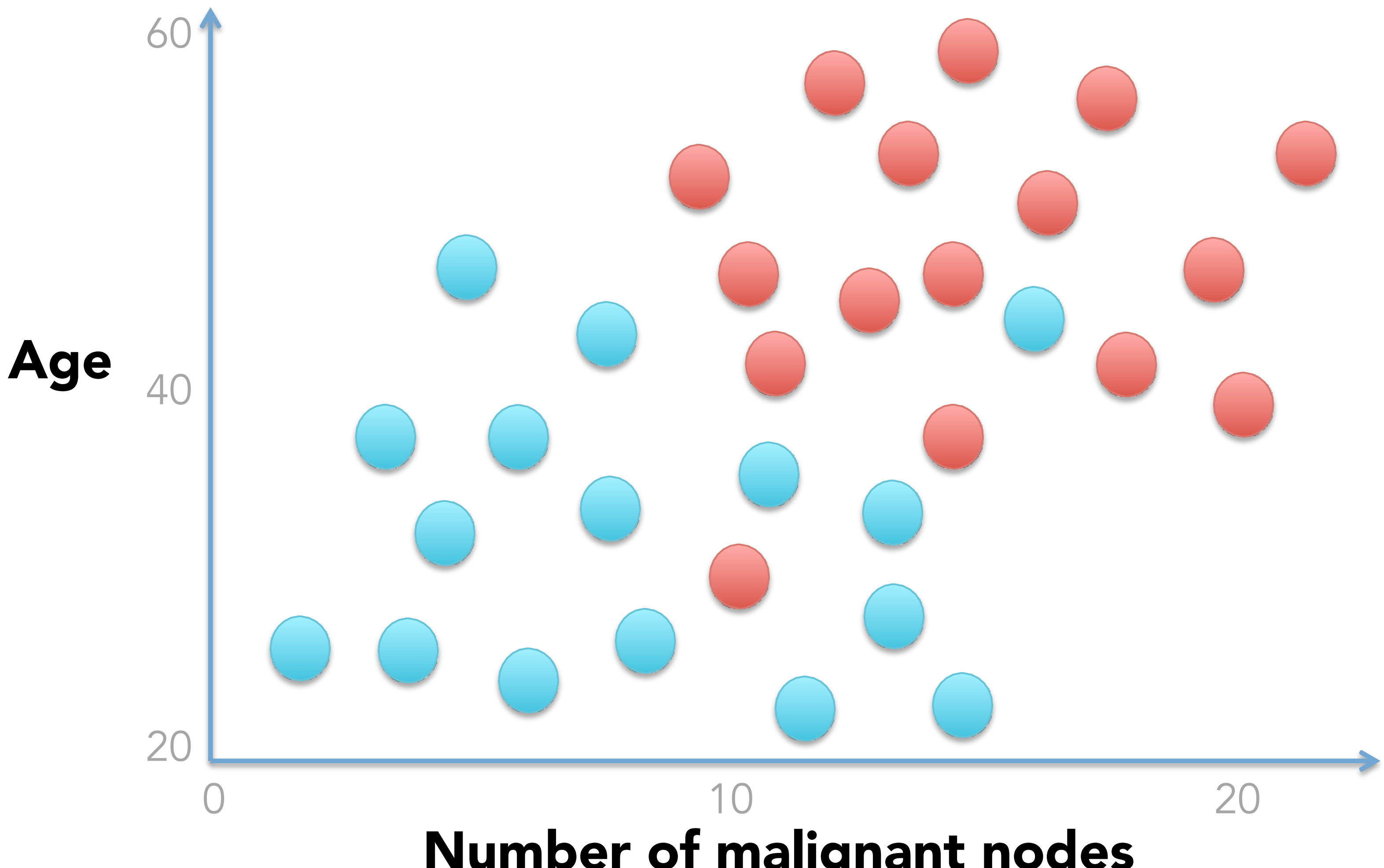
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



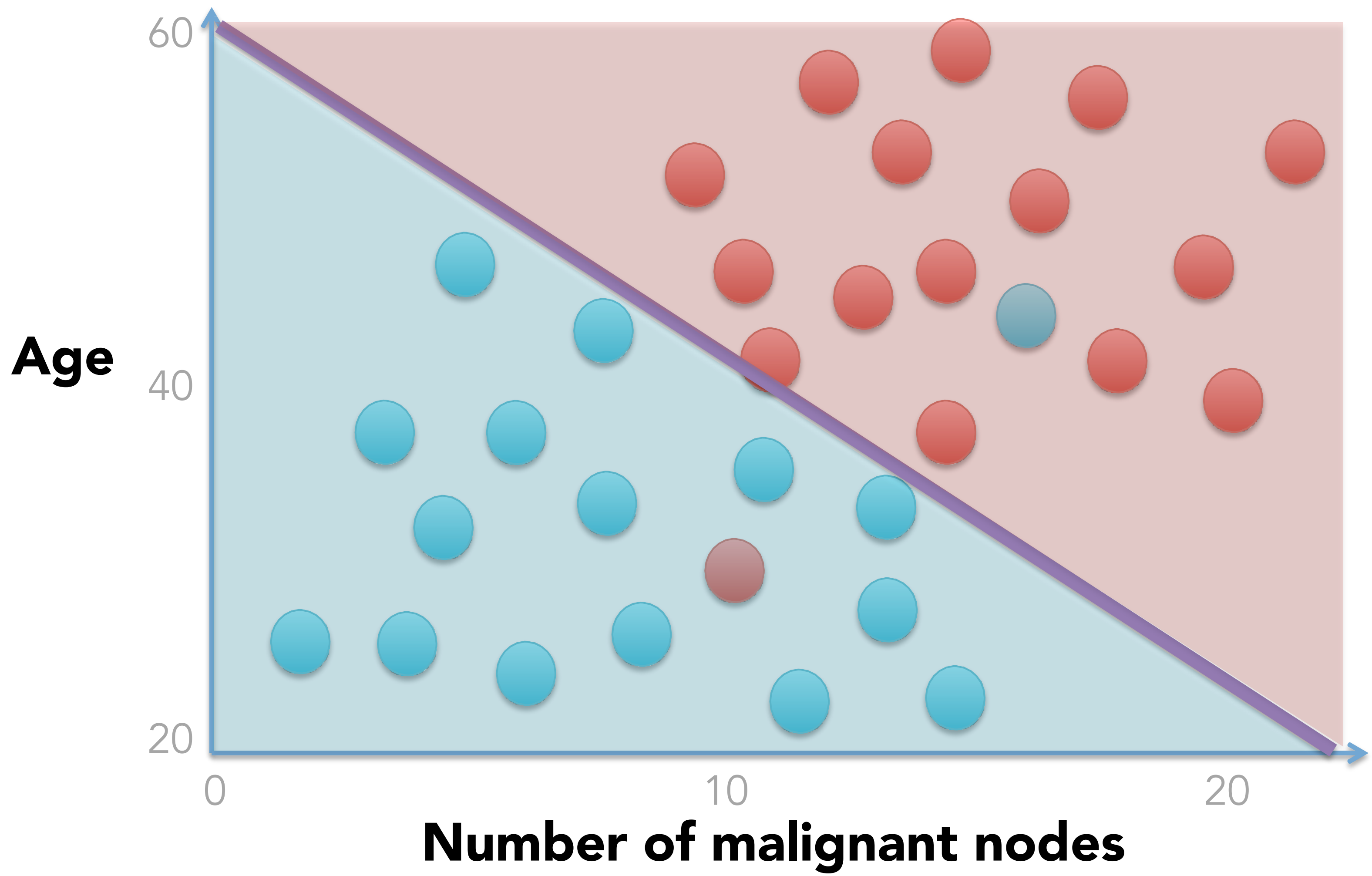
1 Feature. Number of malignant nodes
2 Labels. Survived / Lost



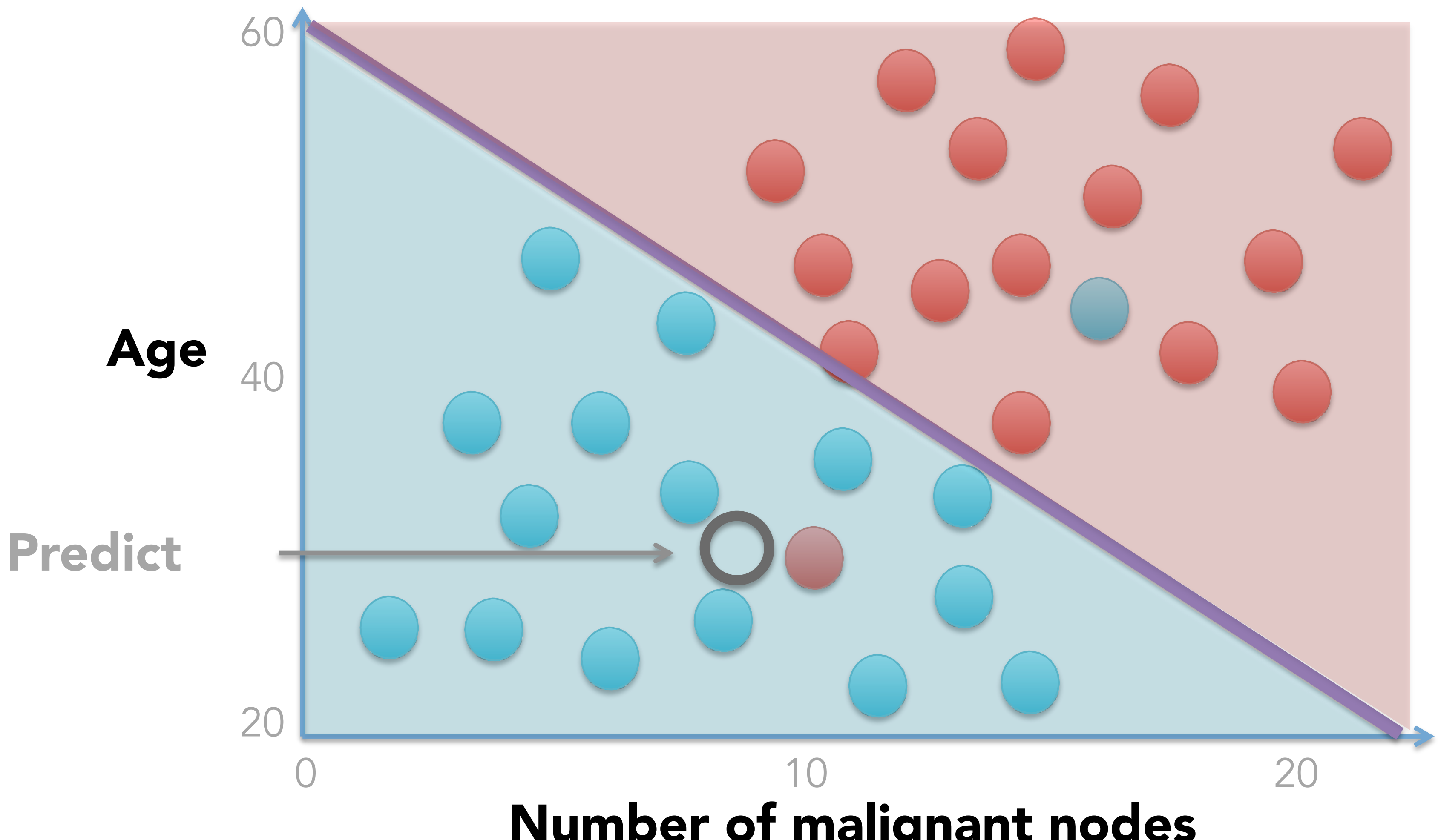
2 Features. No of malignant nodes / Age
2 Labels. Survived / Lost



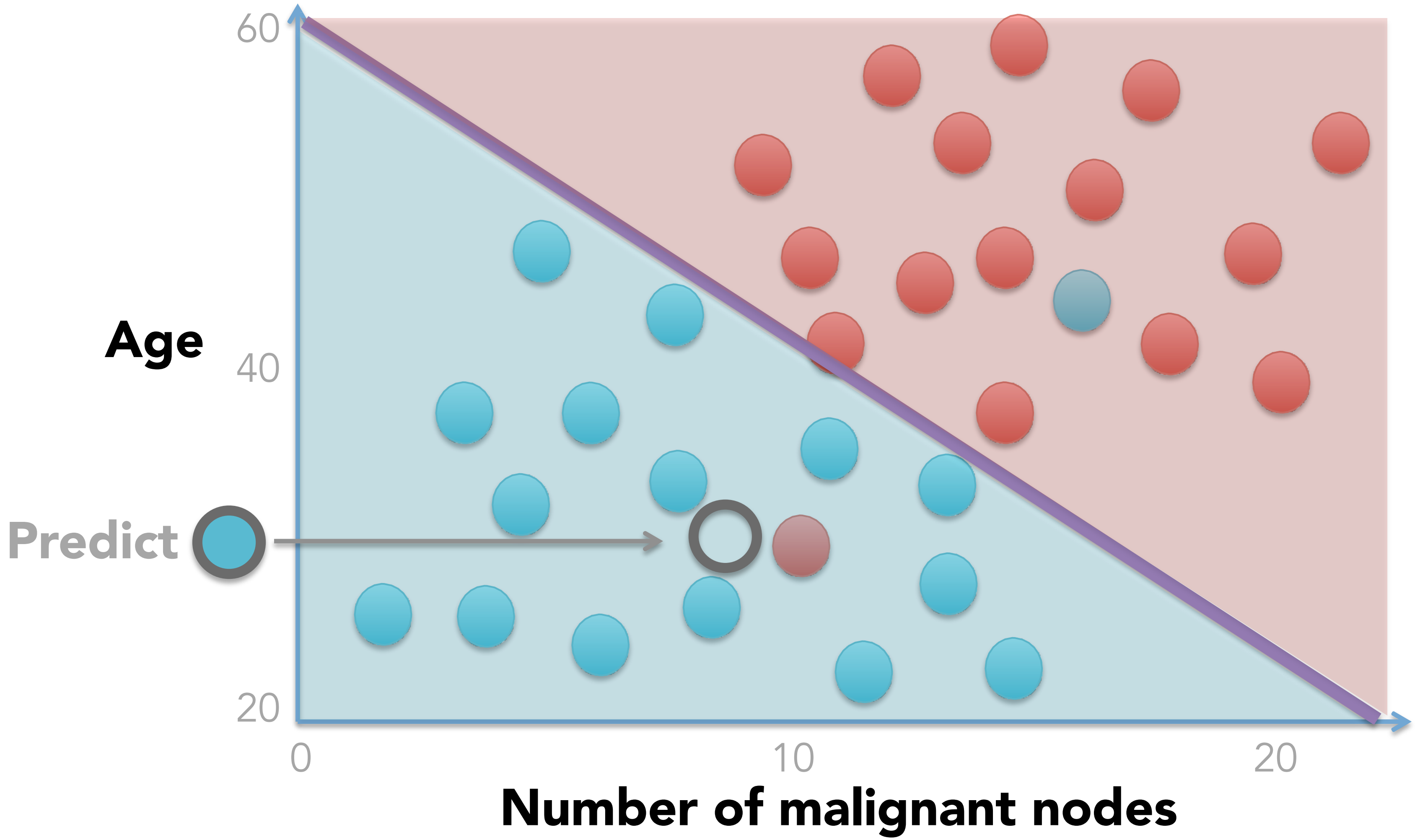
2 Features. No of malignant nodes / Age
2 Labels. Survived / Lost



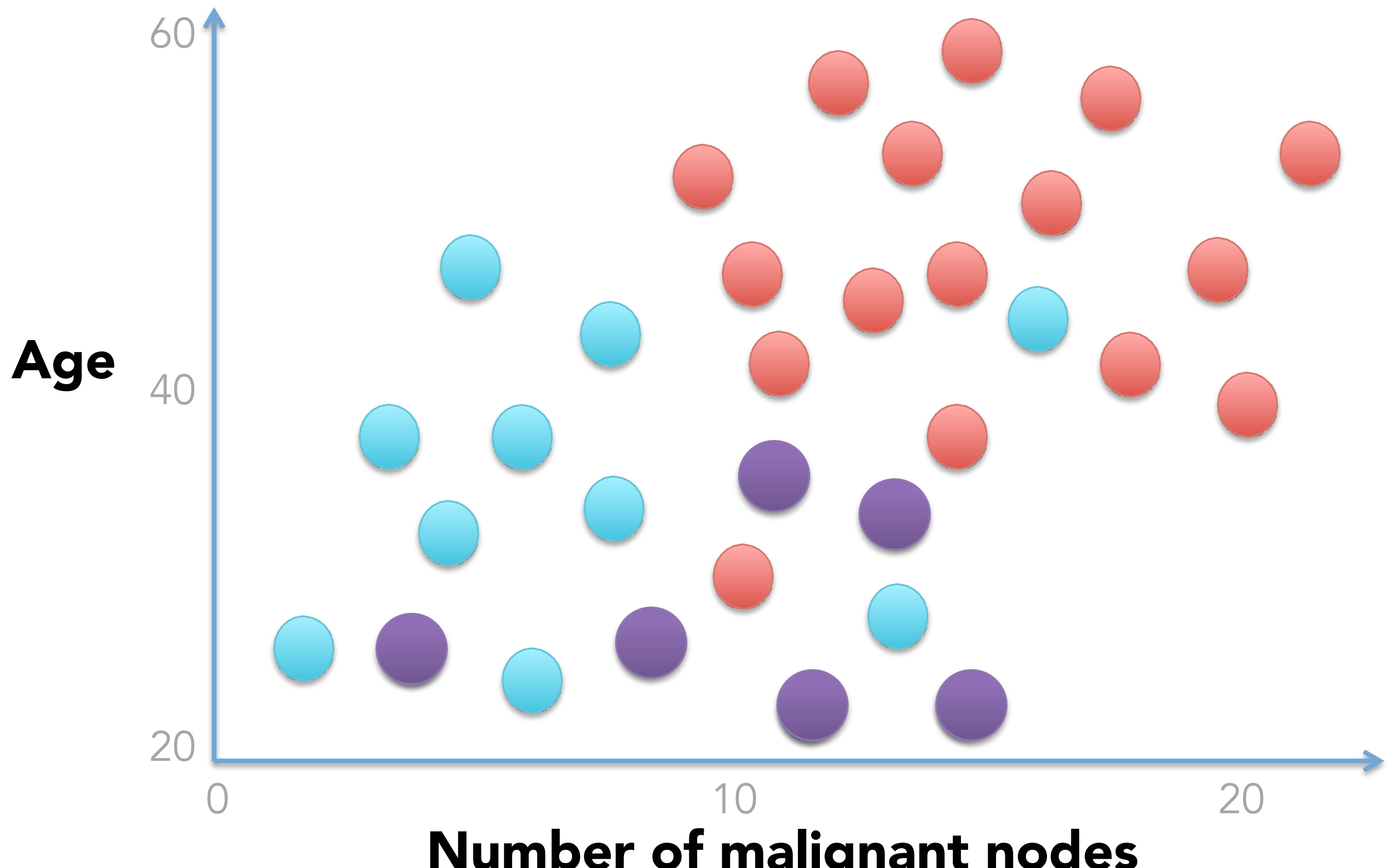
2 Features. No of malignant nodes / Age
2 Labels. Survived / Lost



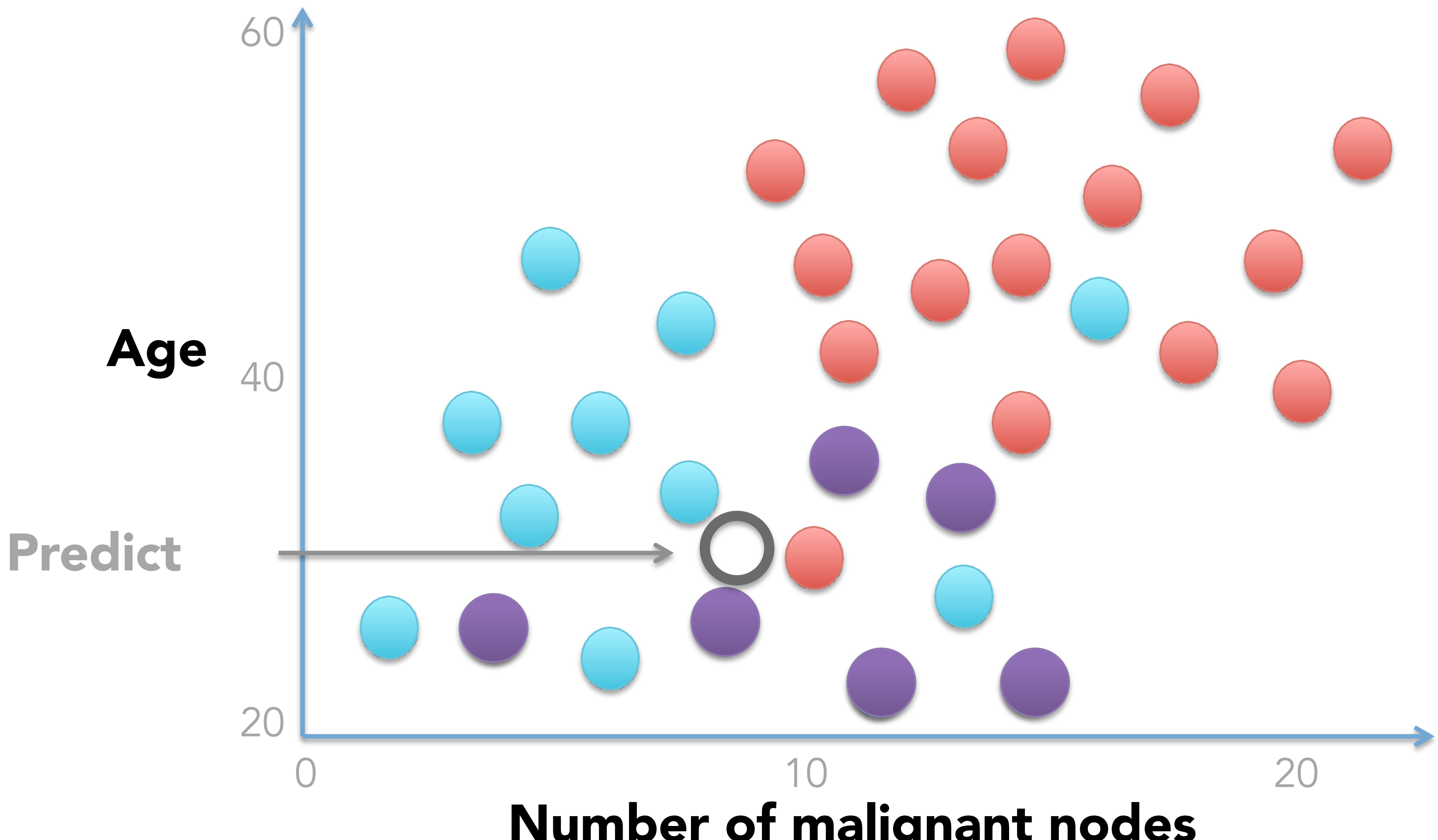
2 Features. No of malignant nodes / Age
2 Labels. Survived / Lost



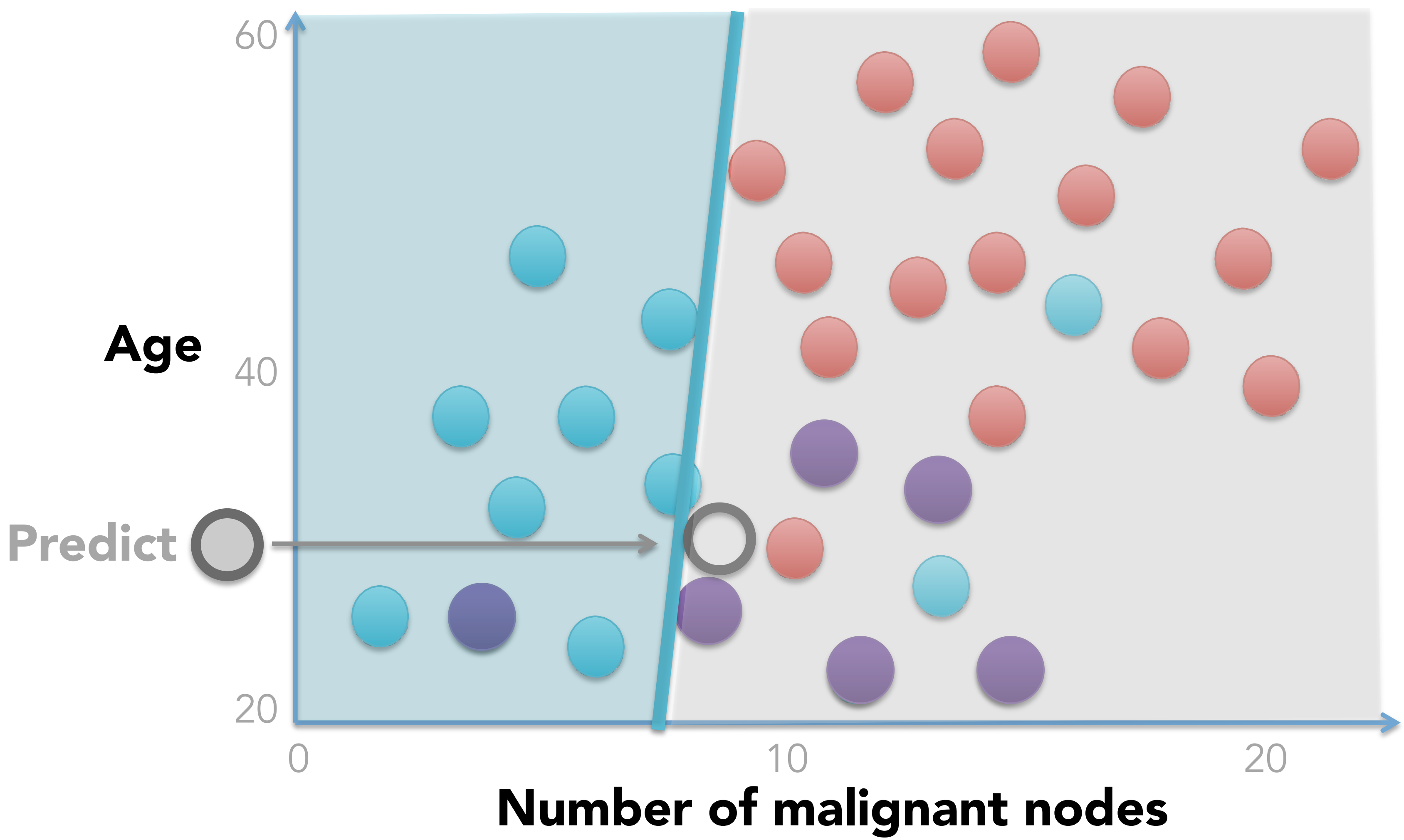
2 Features. No of malignant nodes / Age
3 Labels. Healthy / Complications / Lost



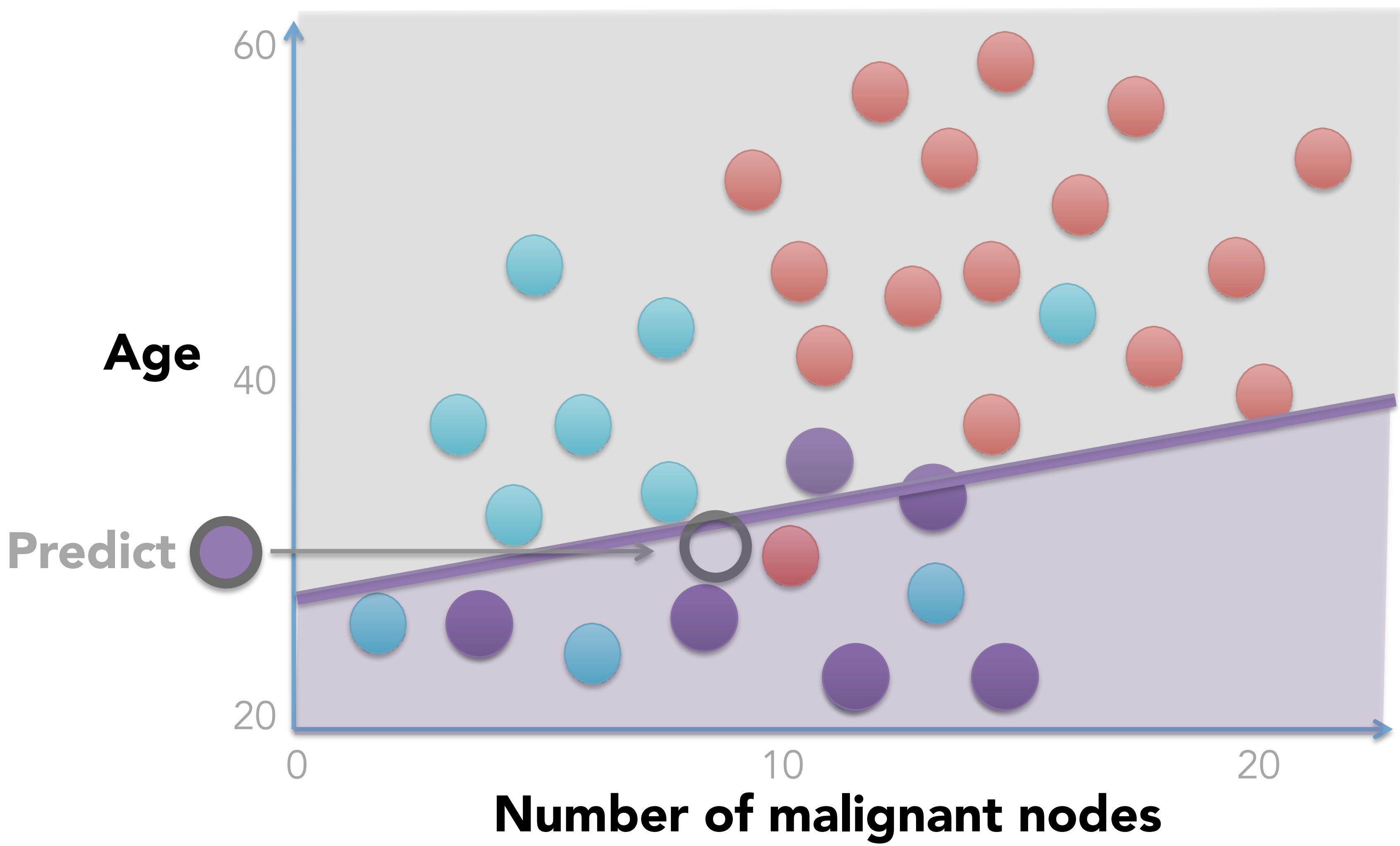
2 Features. No of malignant nodes / Age
3 Labels. Healthy / Complications / Lost



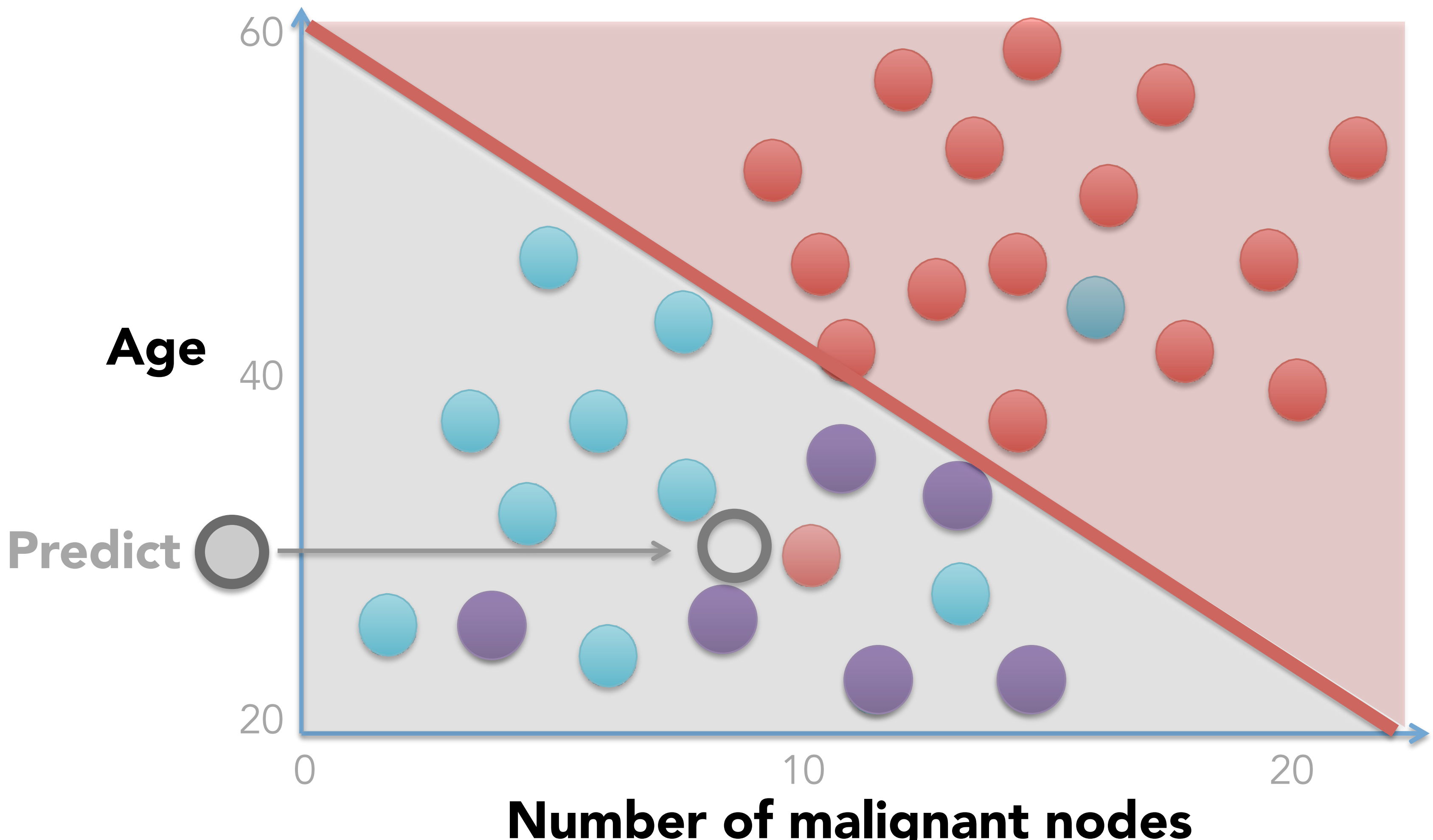
One vs all. Survived vs all.



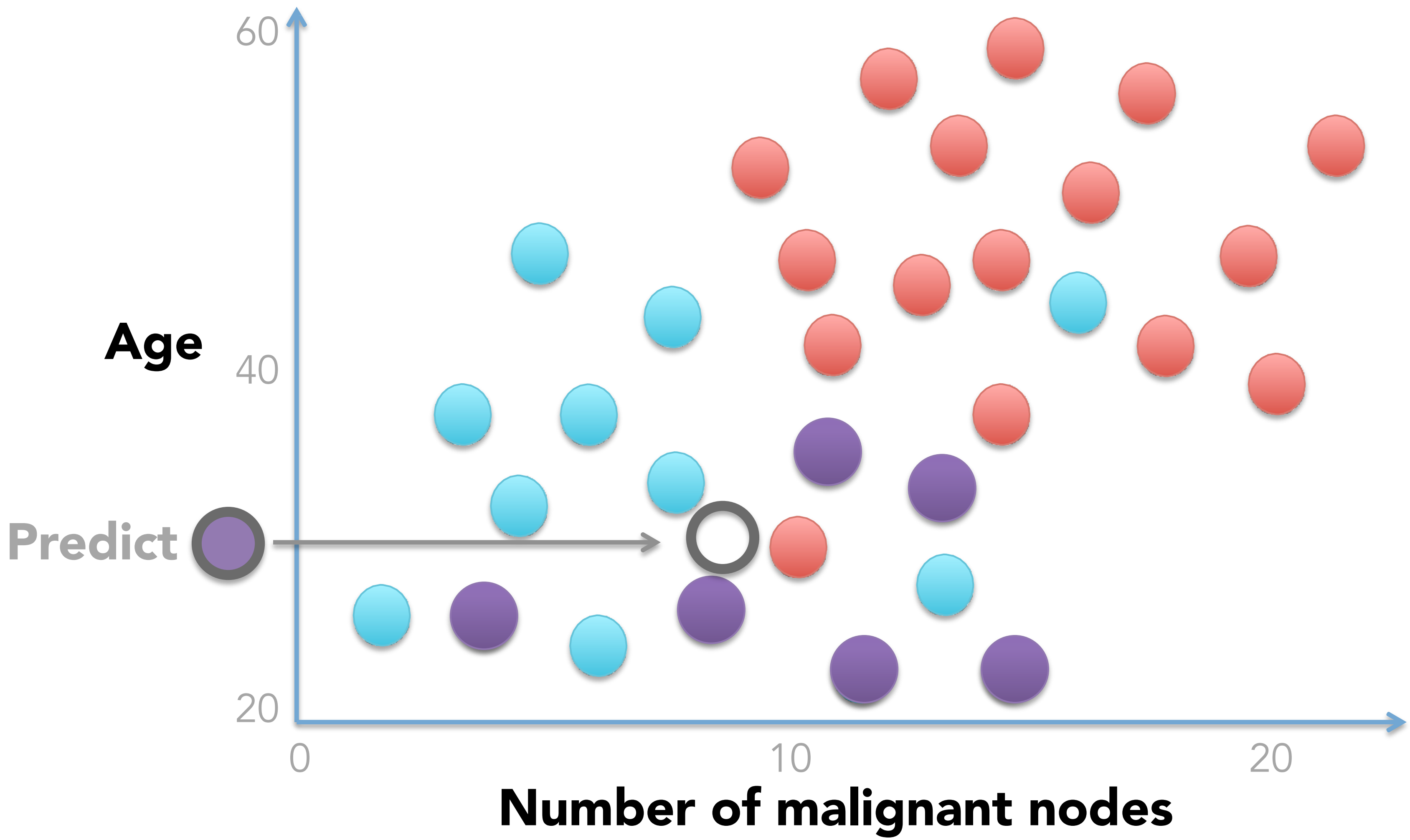
One vs all. Complications vs all.



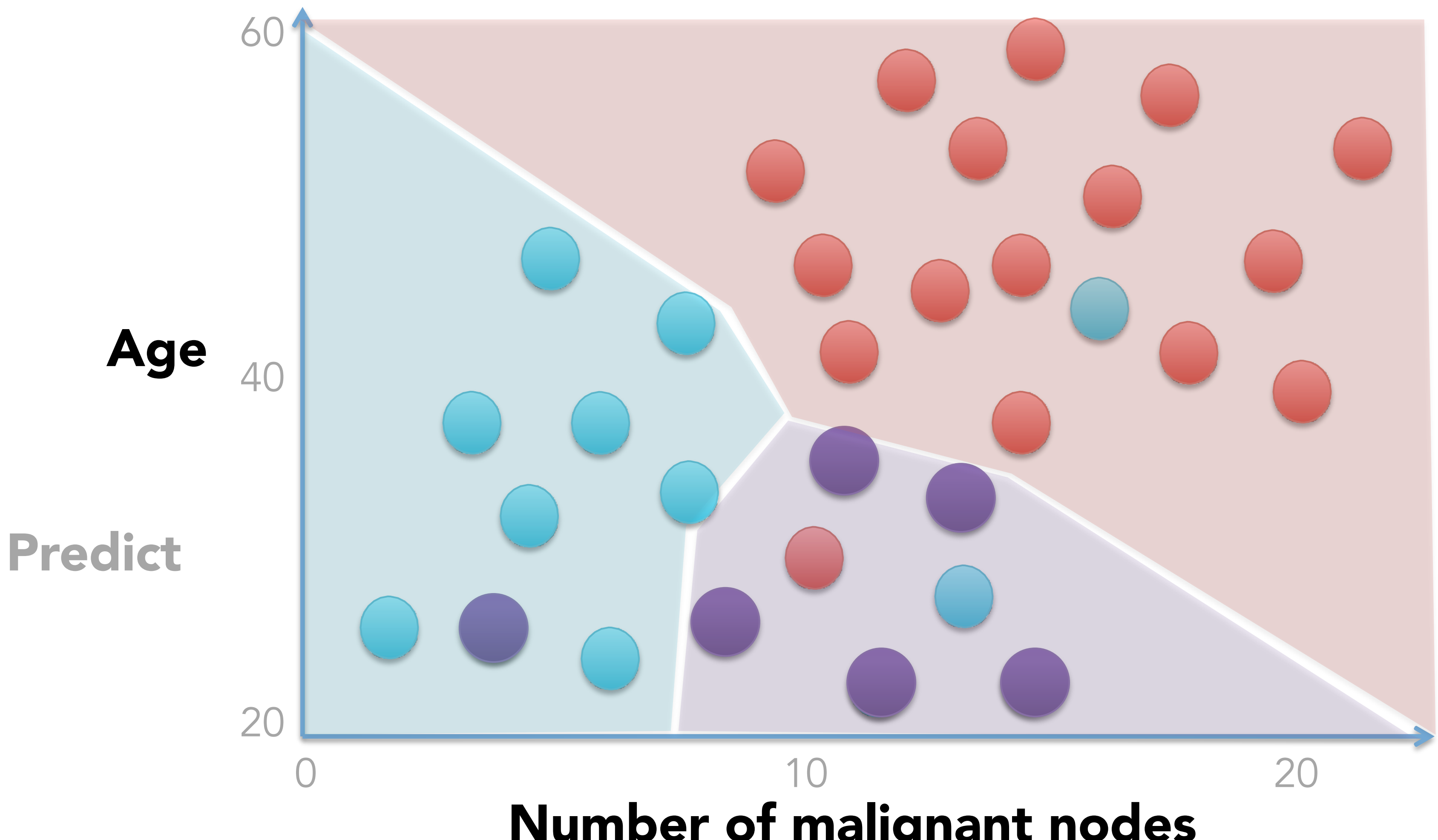
One vs all. Lost vs all.



One vs all. Winner: Complications



One vs all. Essentially, it becomes:



"Logistic regression"
is a classification algorithm

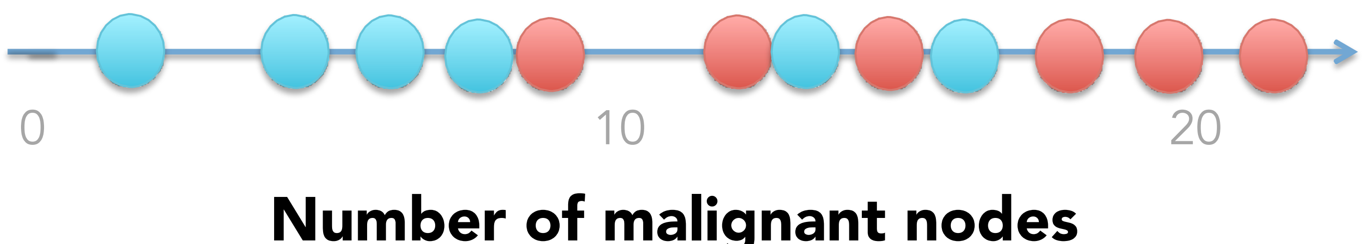
```
from sklearn.linear_model import LogisticRegression
#( just like LinearRegression )
#( Regularization already turned on by default )


from statsmodels.formula.api import Logit
#( just like OLS )
#( For regularization, use .fit_regularized() instead of .fit() )
```

Regression? Pfffsht.

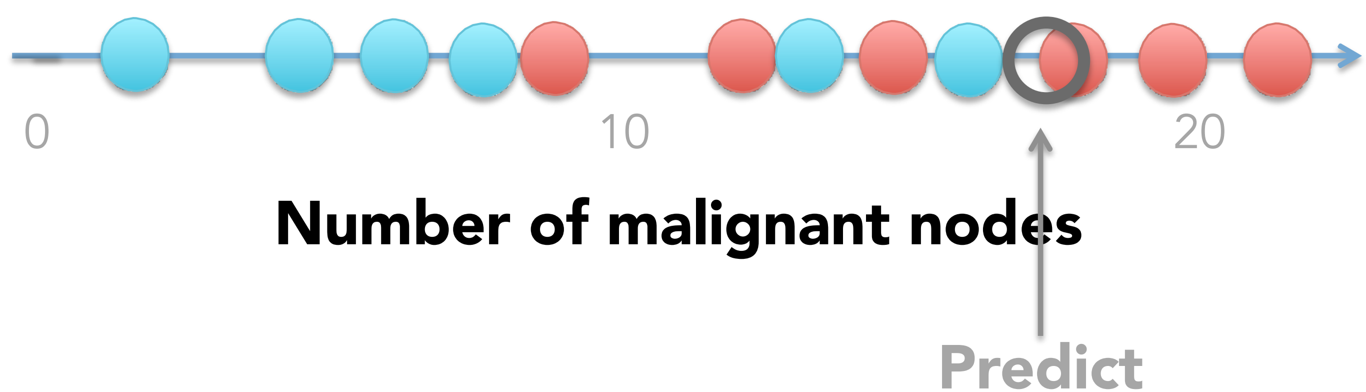
Regression? Pfffsht.

K NEAREST NEIGHBORS

K Nearest Neighbors



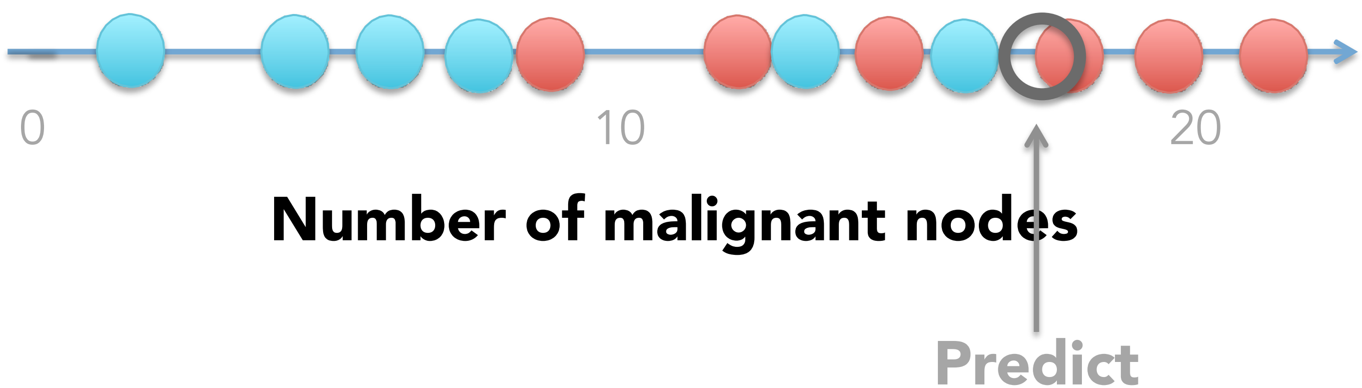
K Nearest Neighbors



K Nearest Neighbors

K=1

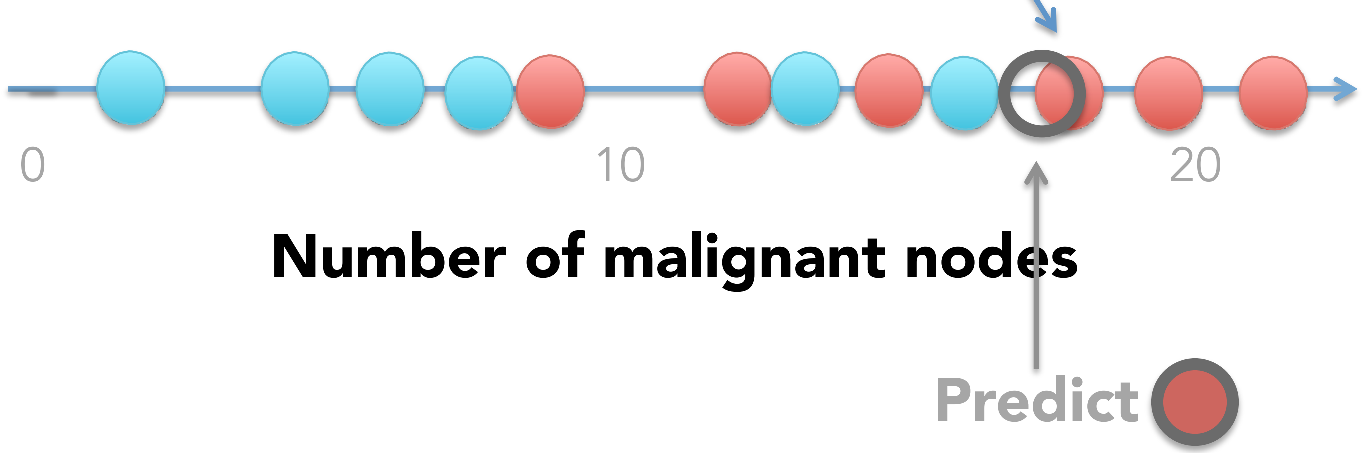
Look at the nearest neighbor, predict their label



K Nearest Neighbors

K=1

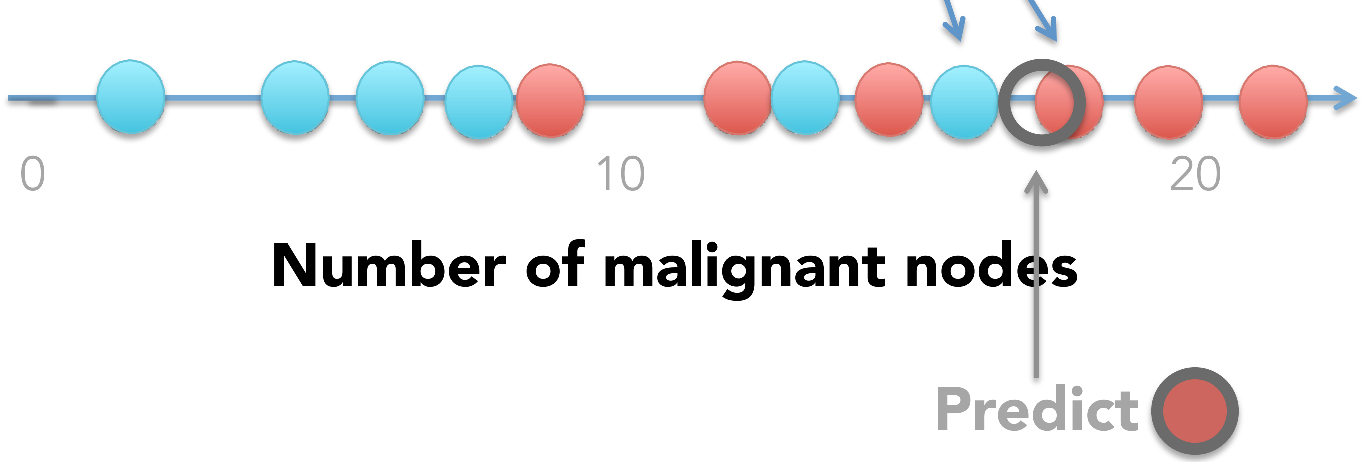
Look at the **nearest neighbor**, predict their label



K Nearest Neighbors

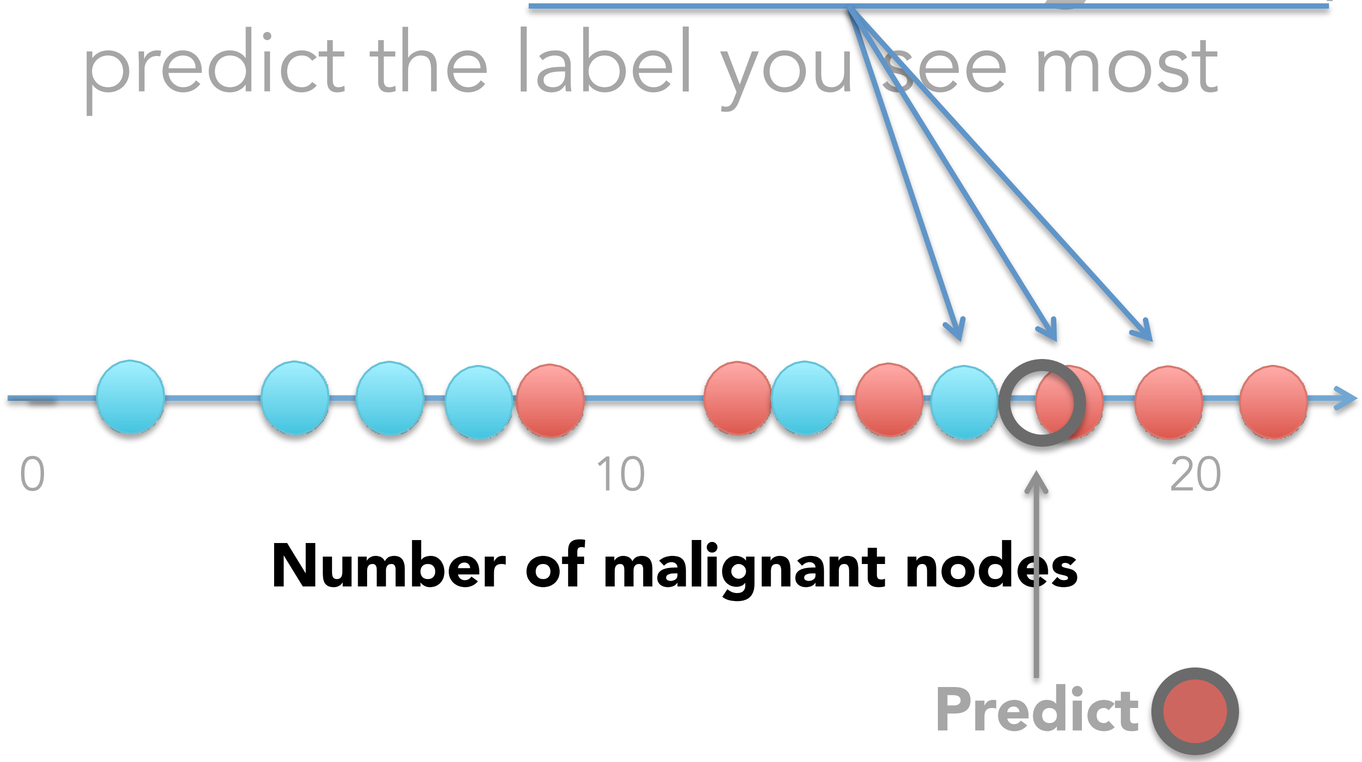
K=2

Look at the **2 nearest neighbors**, predict the label you see most



K Nearest Neighbors

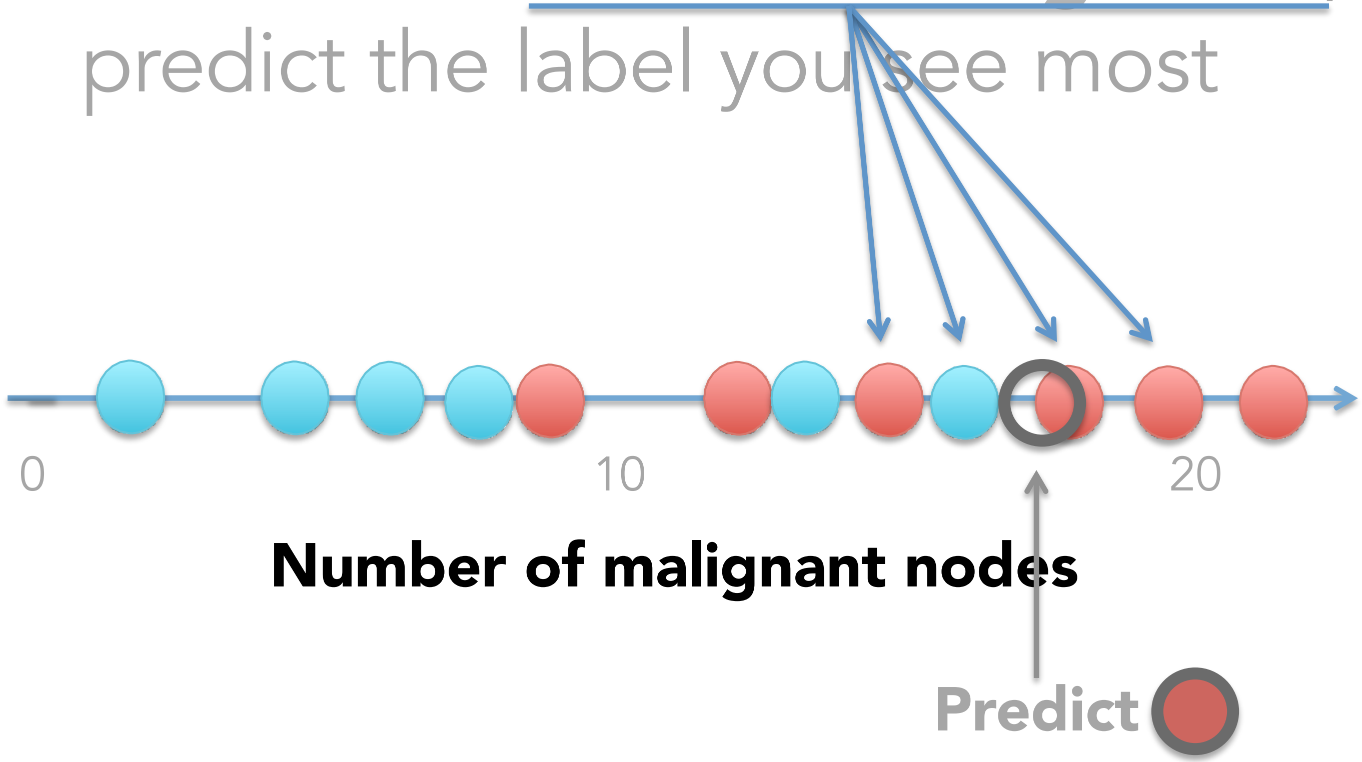
K=3 Look at the **3 nearest neighbors**, predict the label you see most



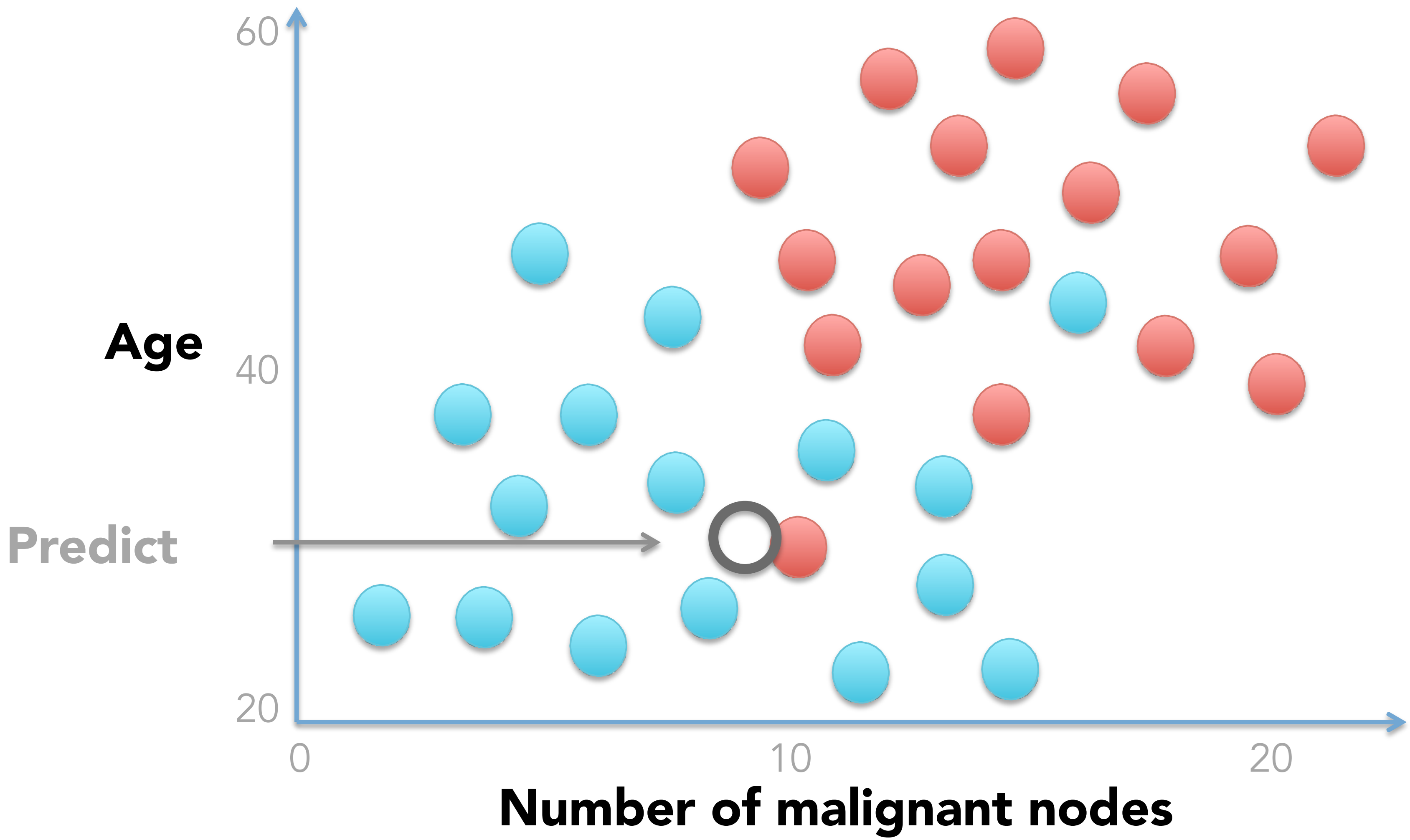
K Nearest Neighbors

K=4

Look at the **4 nearest neighbors**, predict the label you see most

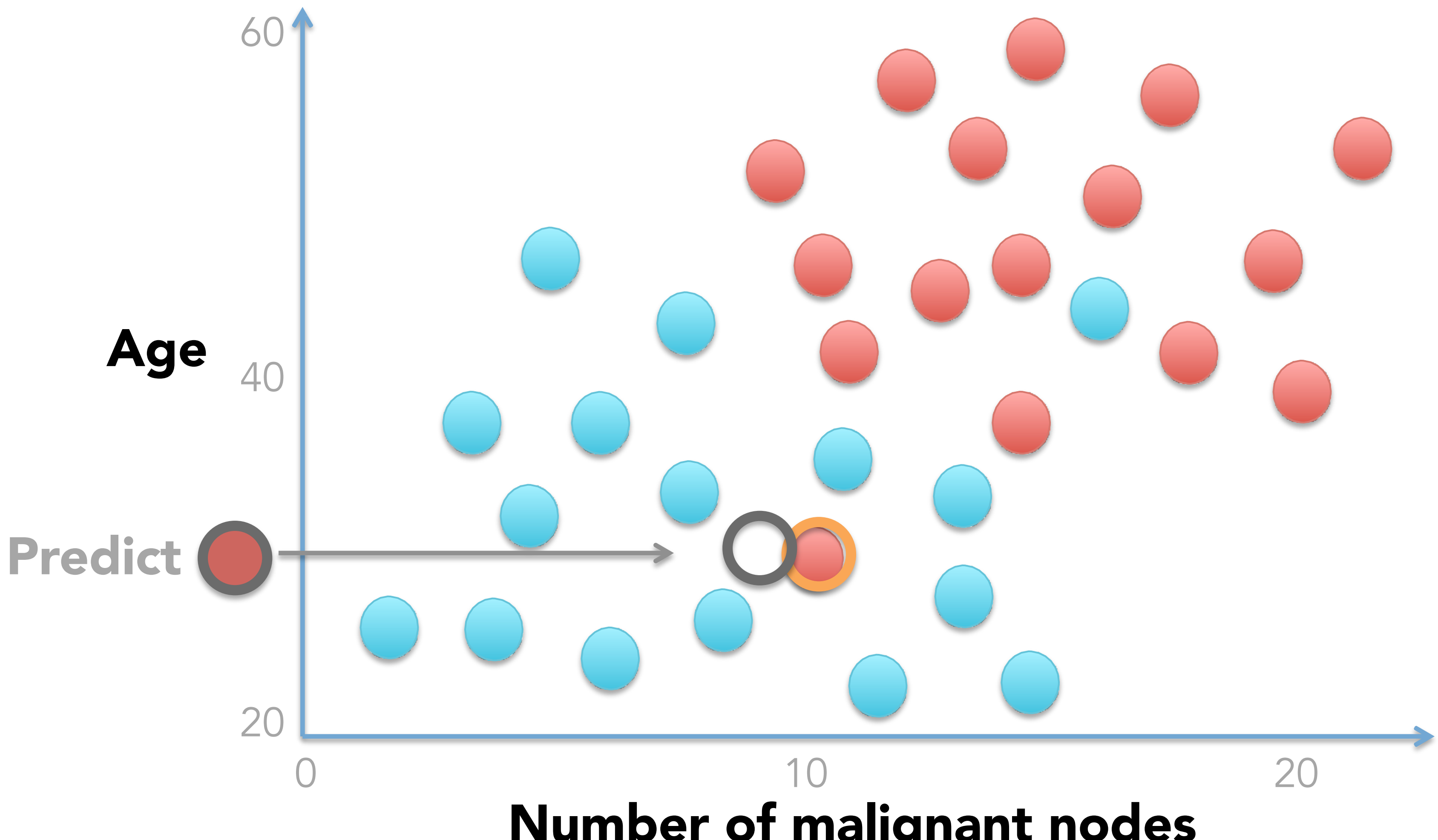


K Nearest Neighbors



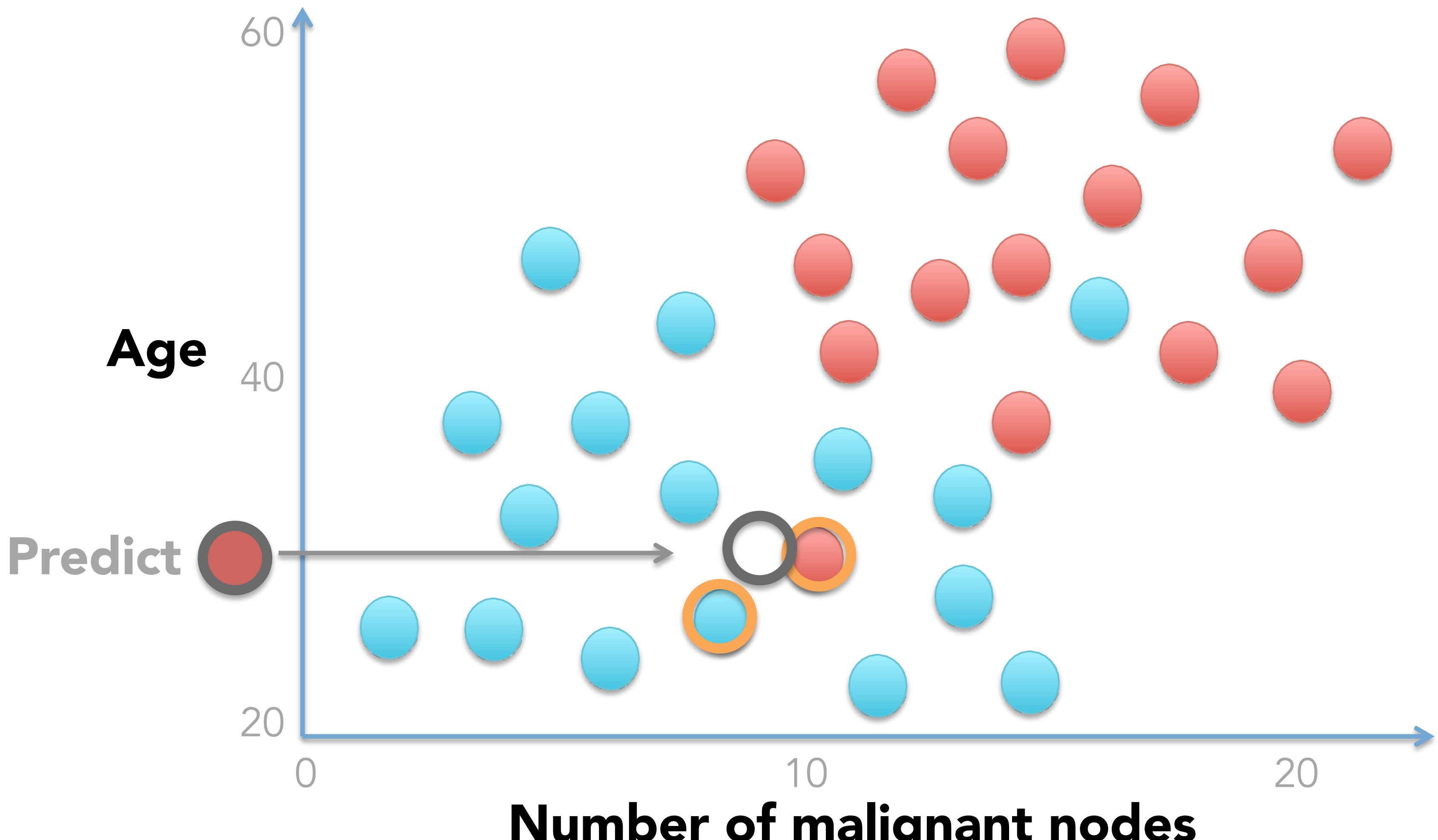
K Nearest Neighbors K=1

Neighbor count: 0 1



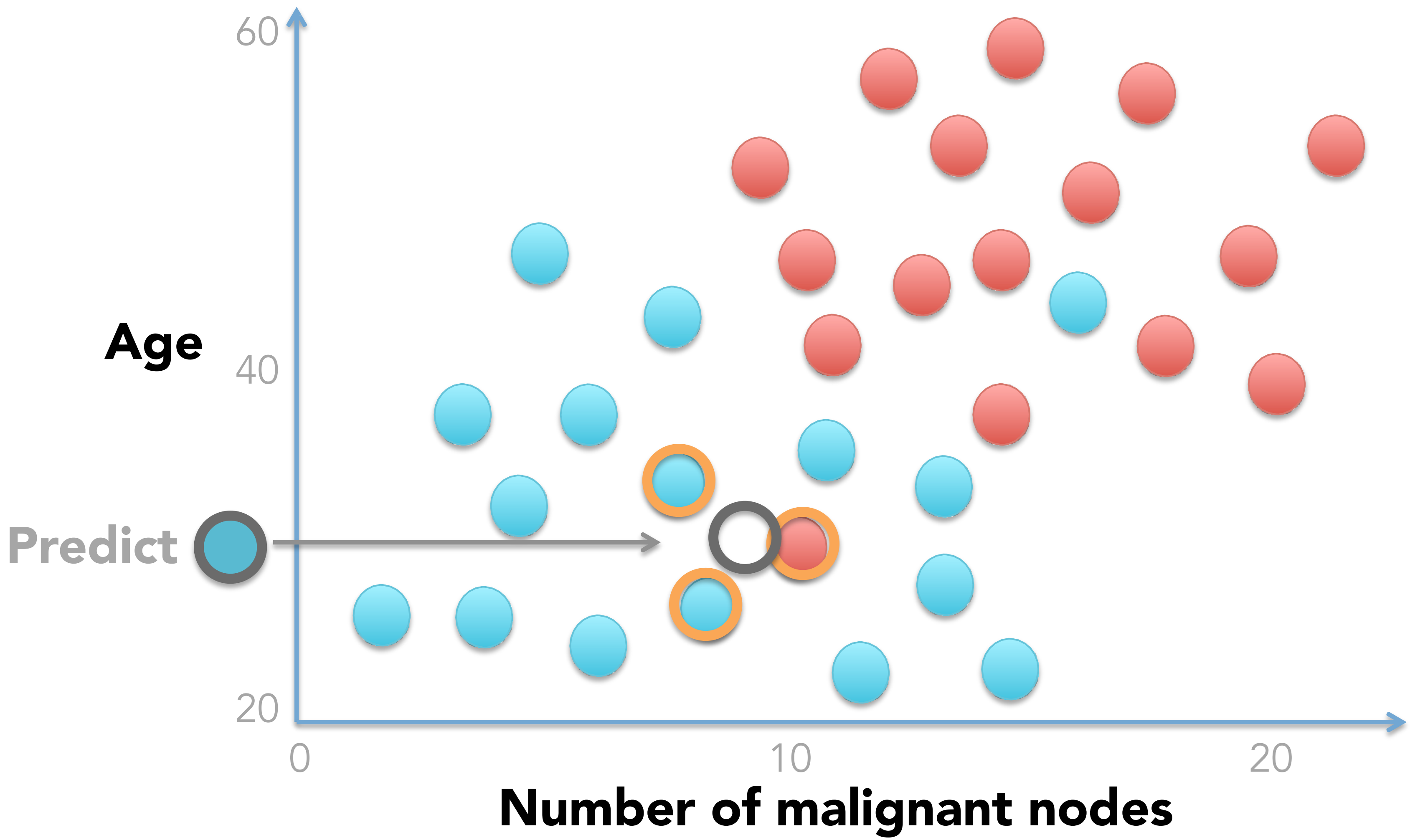
K Nearest Neighbors K=2

Neighbor count: 1 1



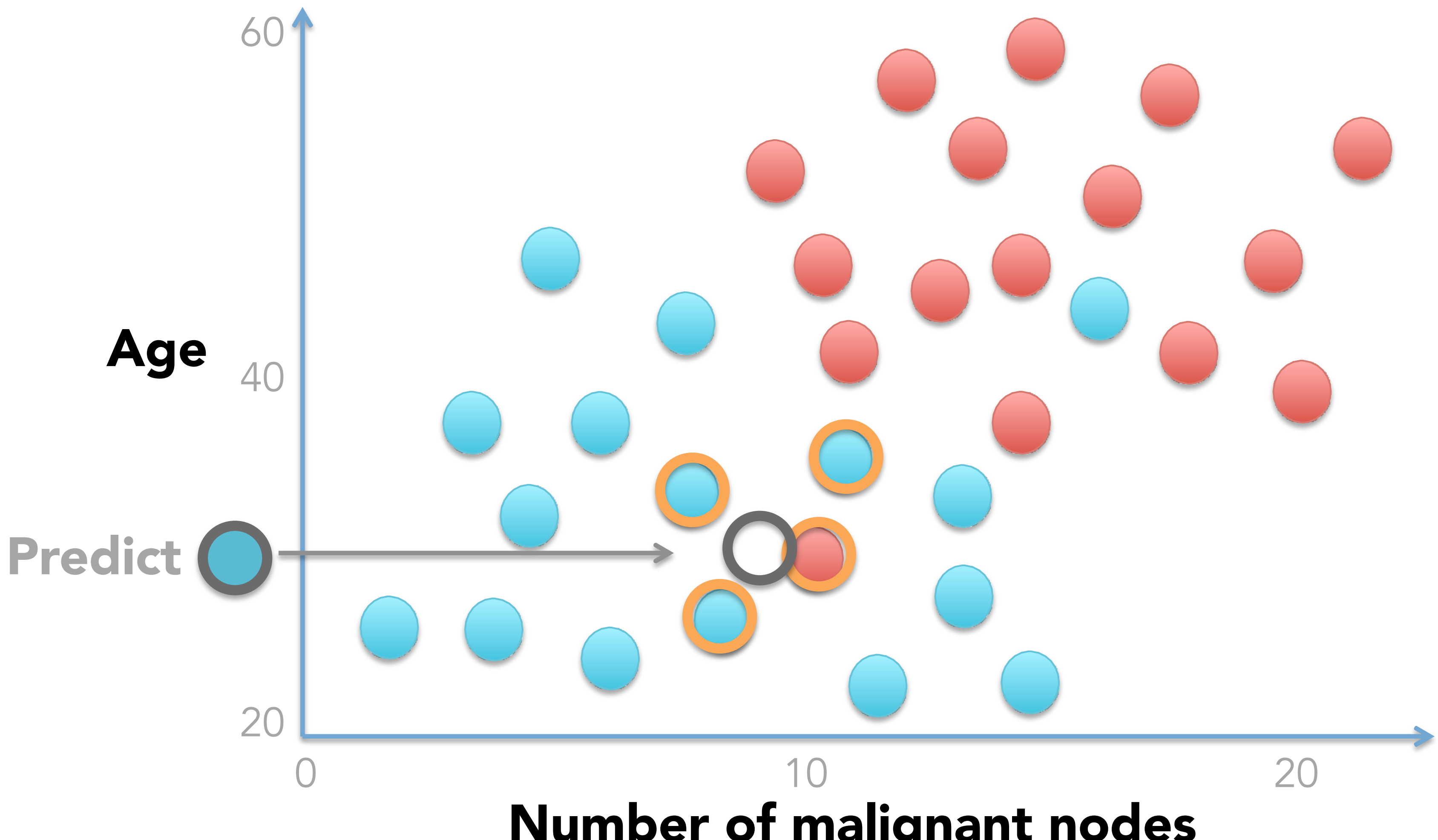
K Nearest Neighbors K=3

Neighbor count: 2 1



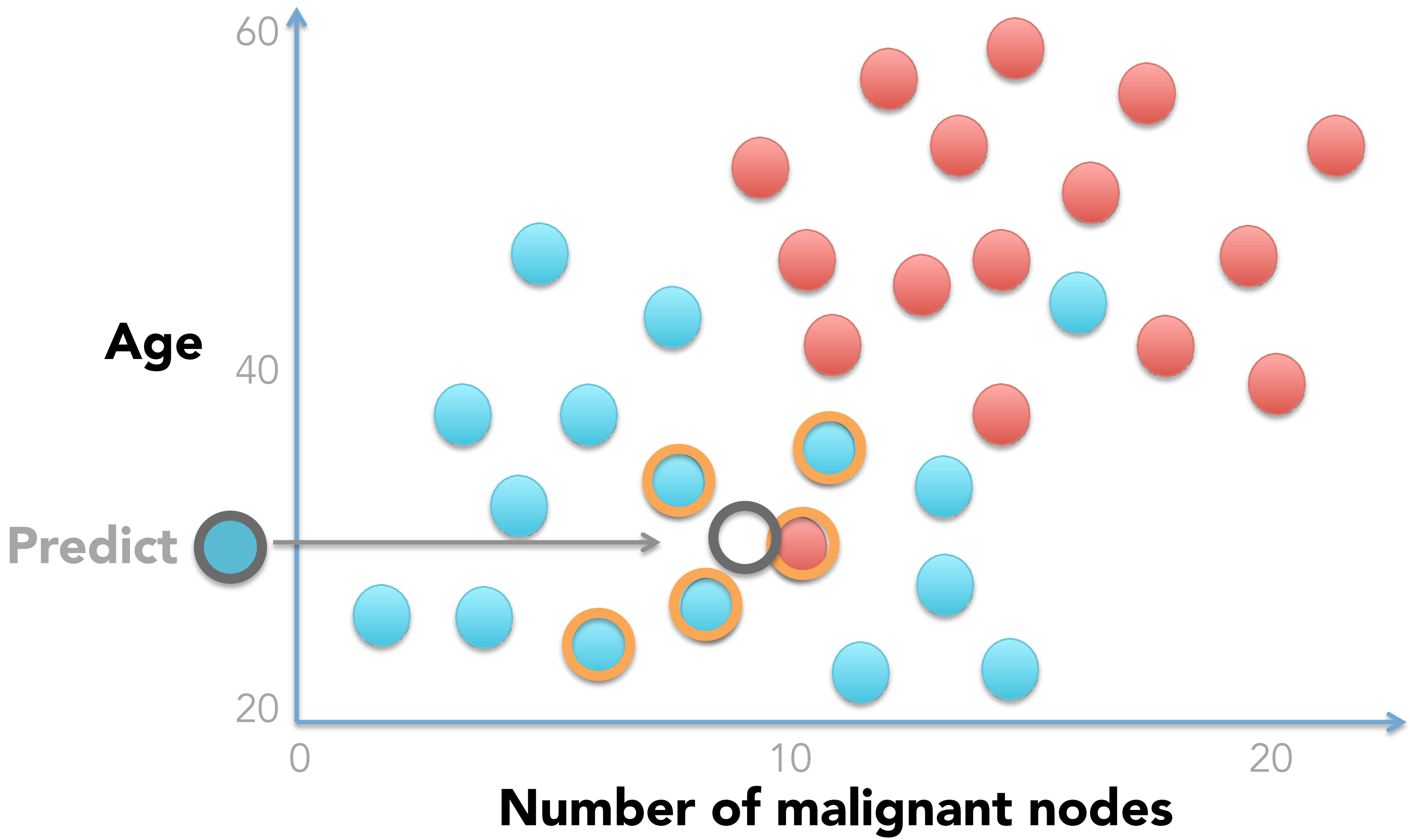
K Nearest Neighbors K=4

Neighbor count: 3 1



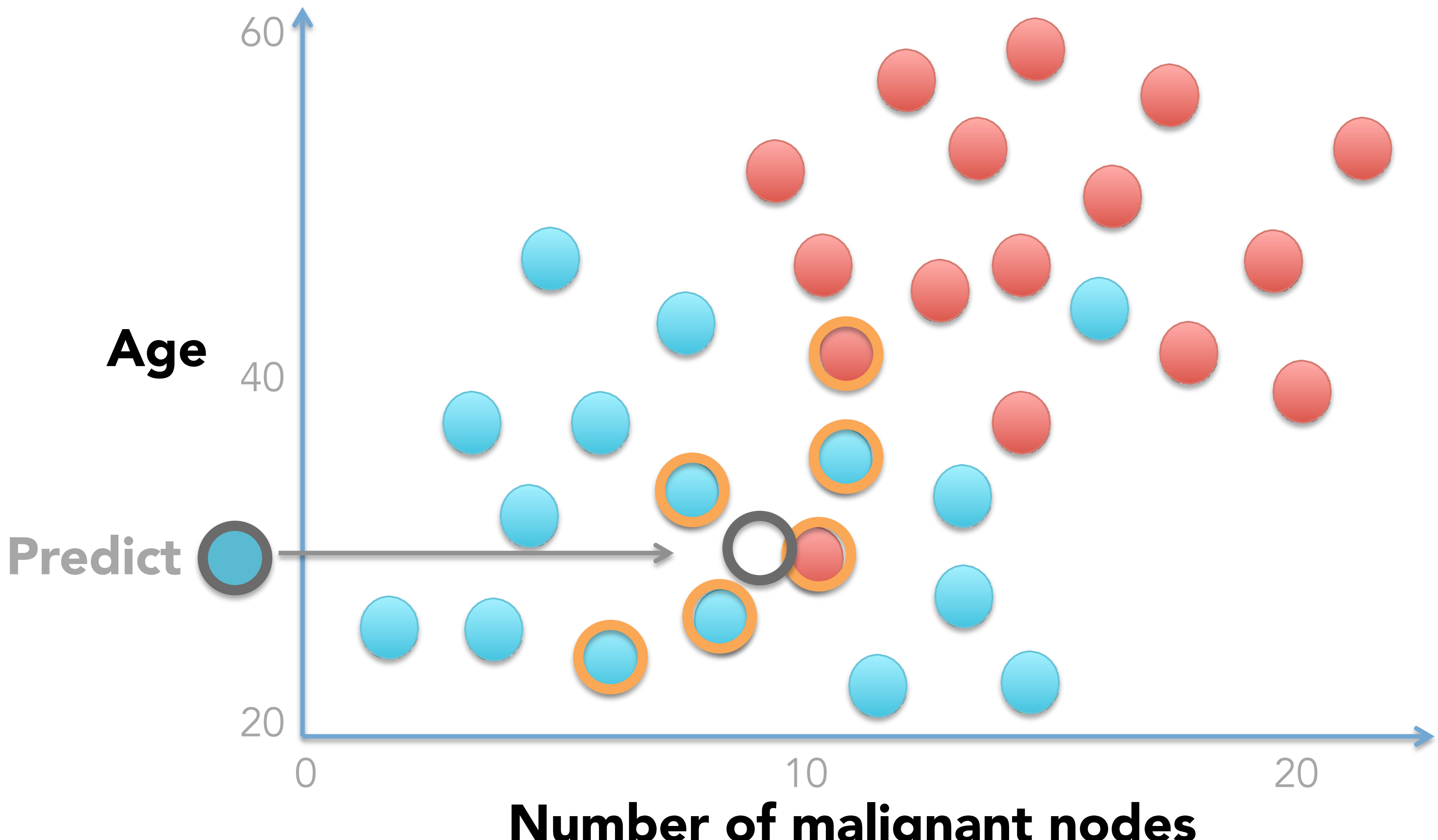
K Nearest Neighbors K=5

Neighbor count: 4 1

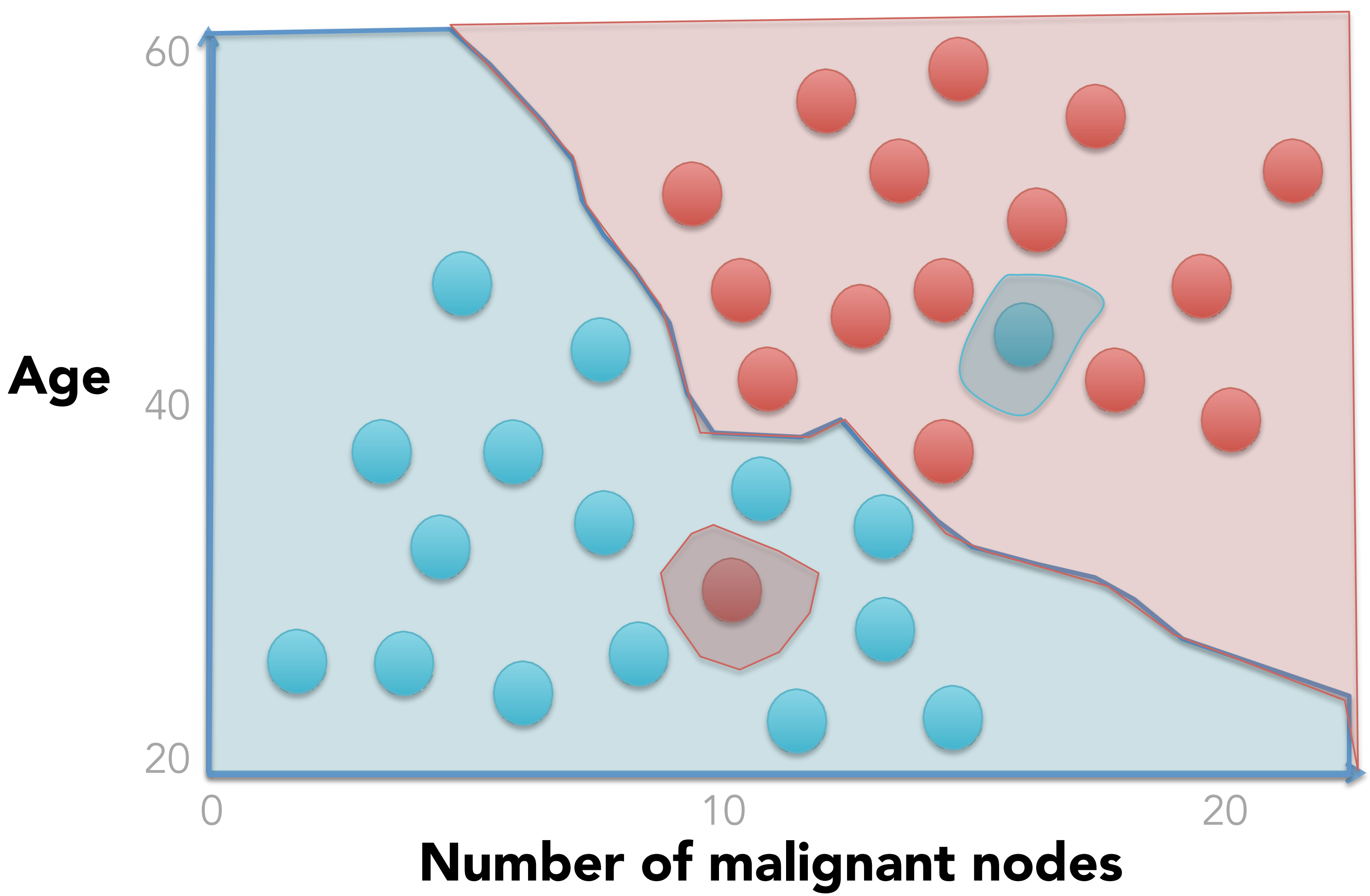


K Nearest Neighbors K=6

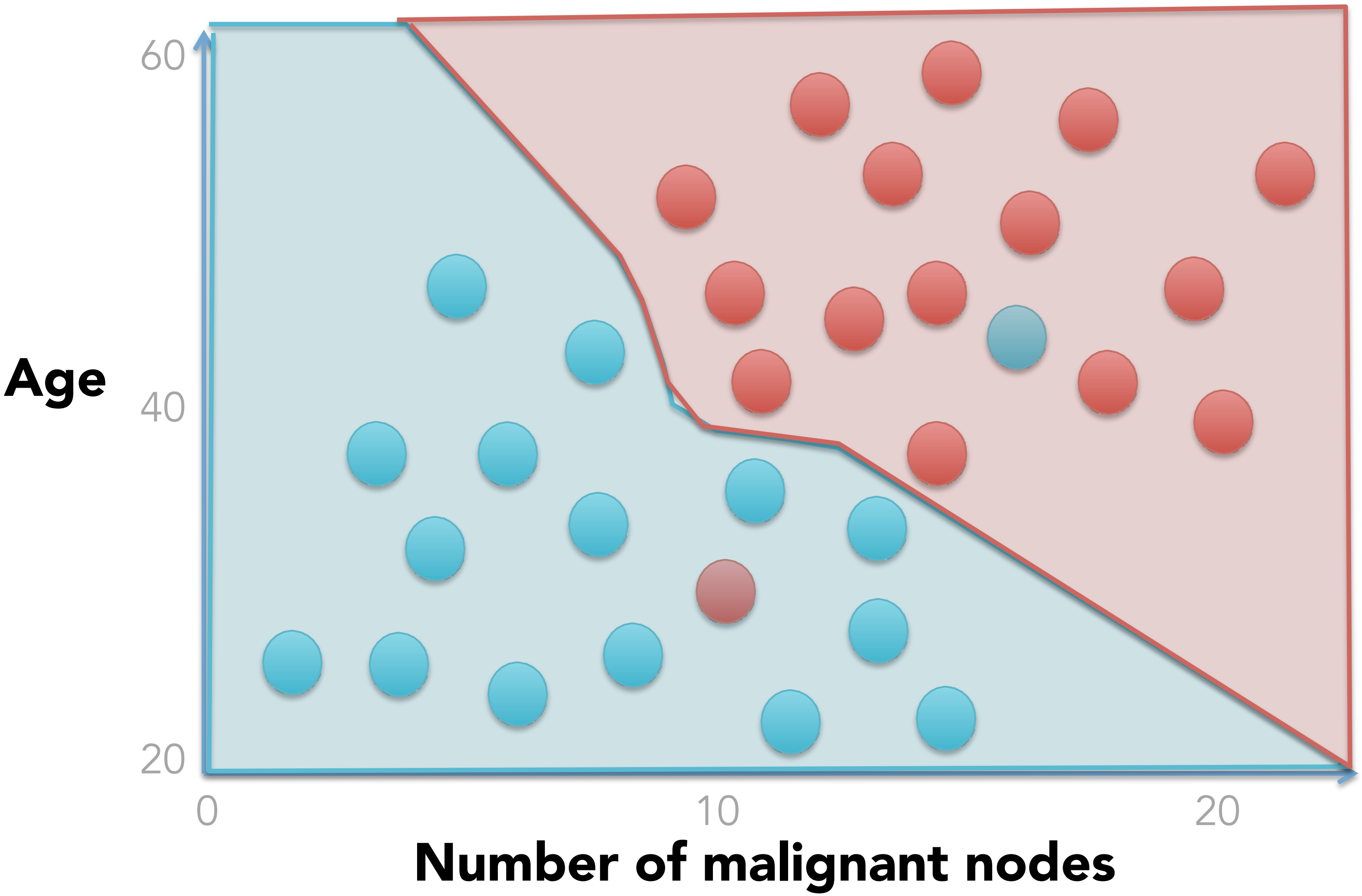
Neighbor count: 4 2



KNN Decision Boundary K=1

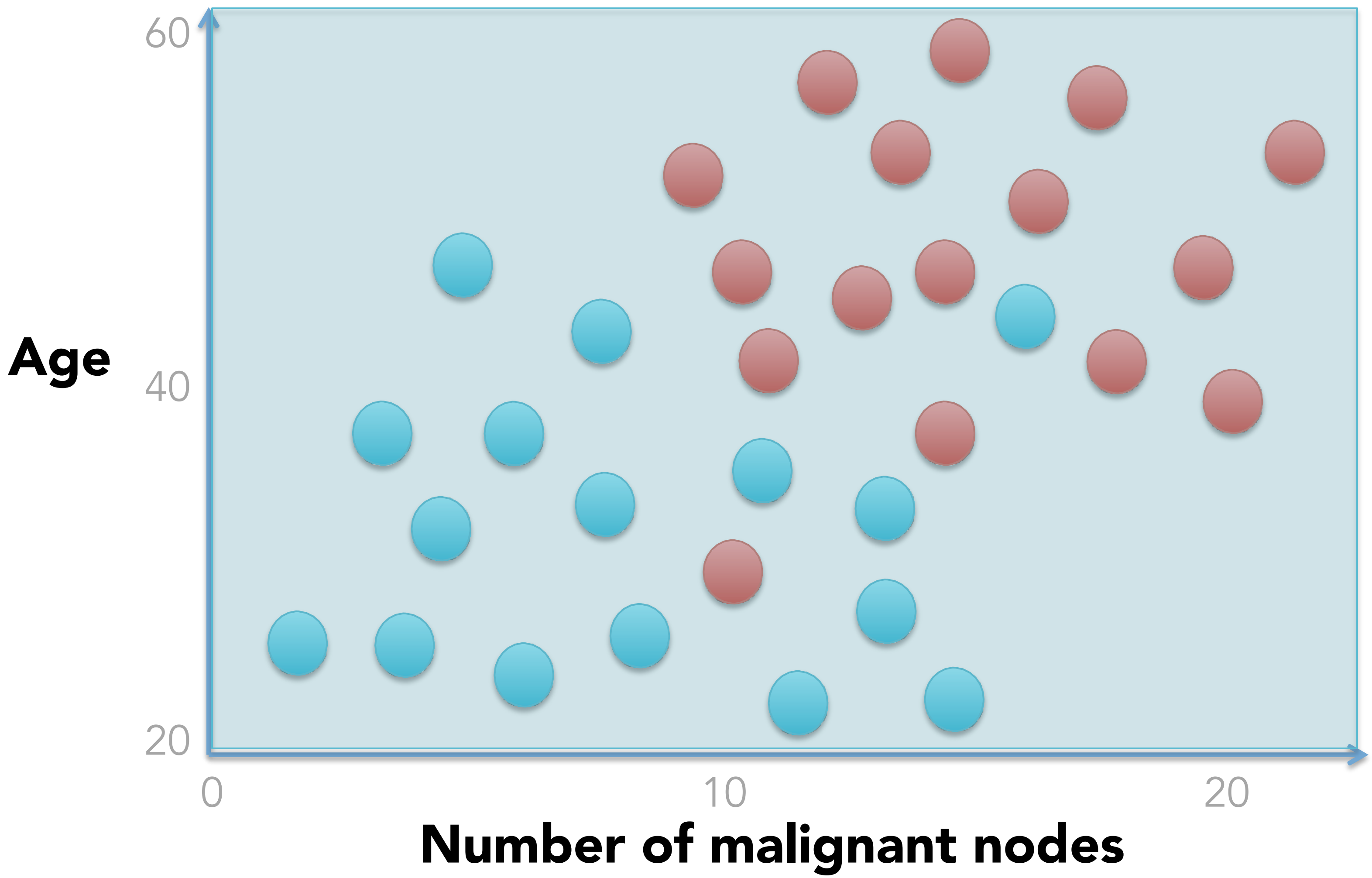


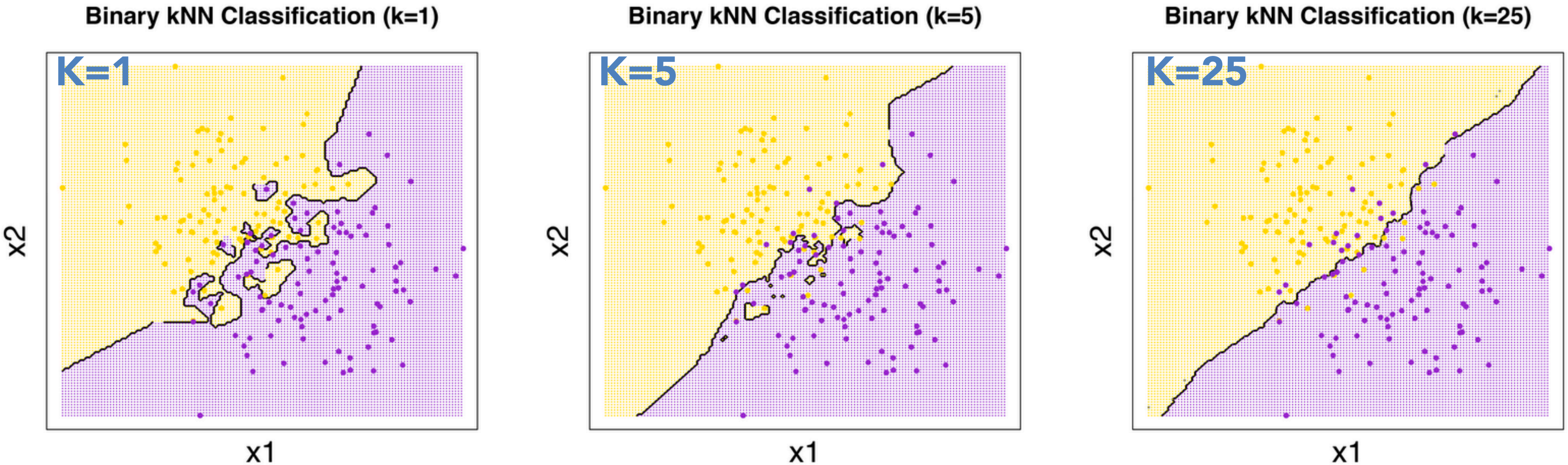
KNN Decision Boundary K=5

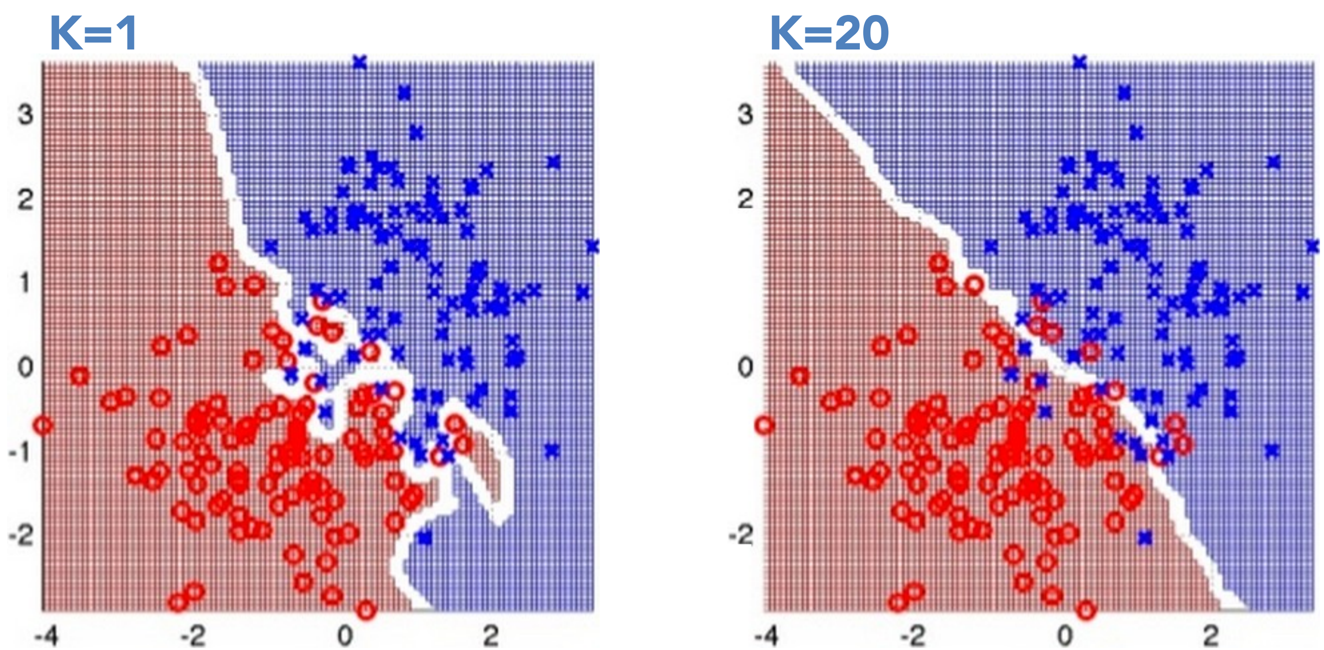


KNN Decision Boundary K=34

Total counts: 18 16



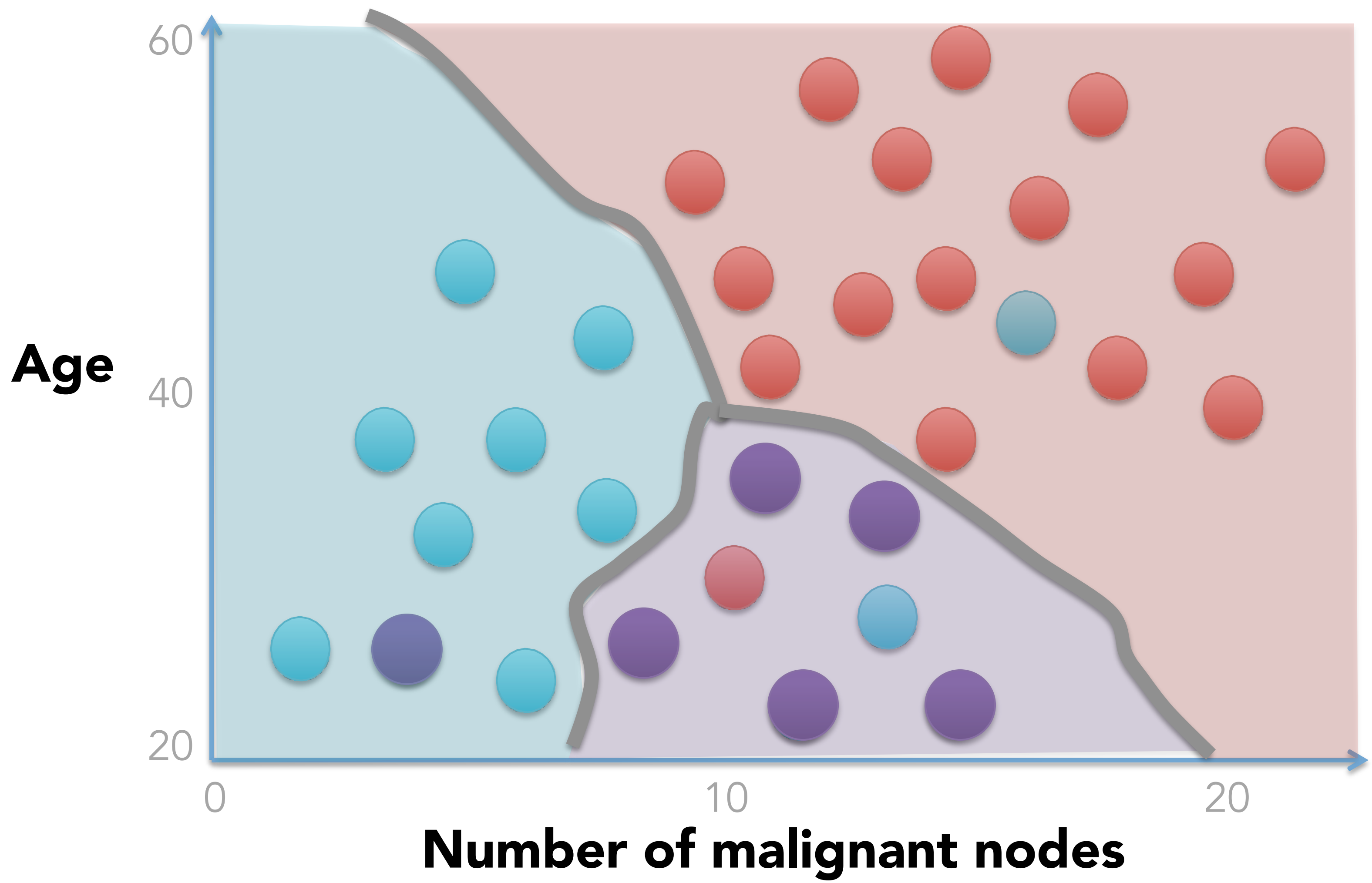




http://bdewilde.github.io/blog/blogger/2012/10/26/classification-of-hand-written-digits-3/

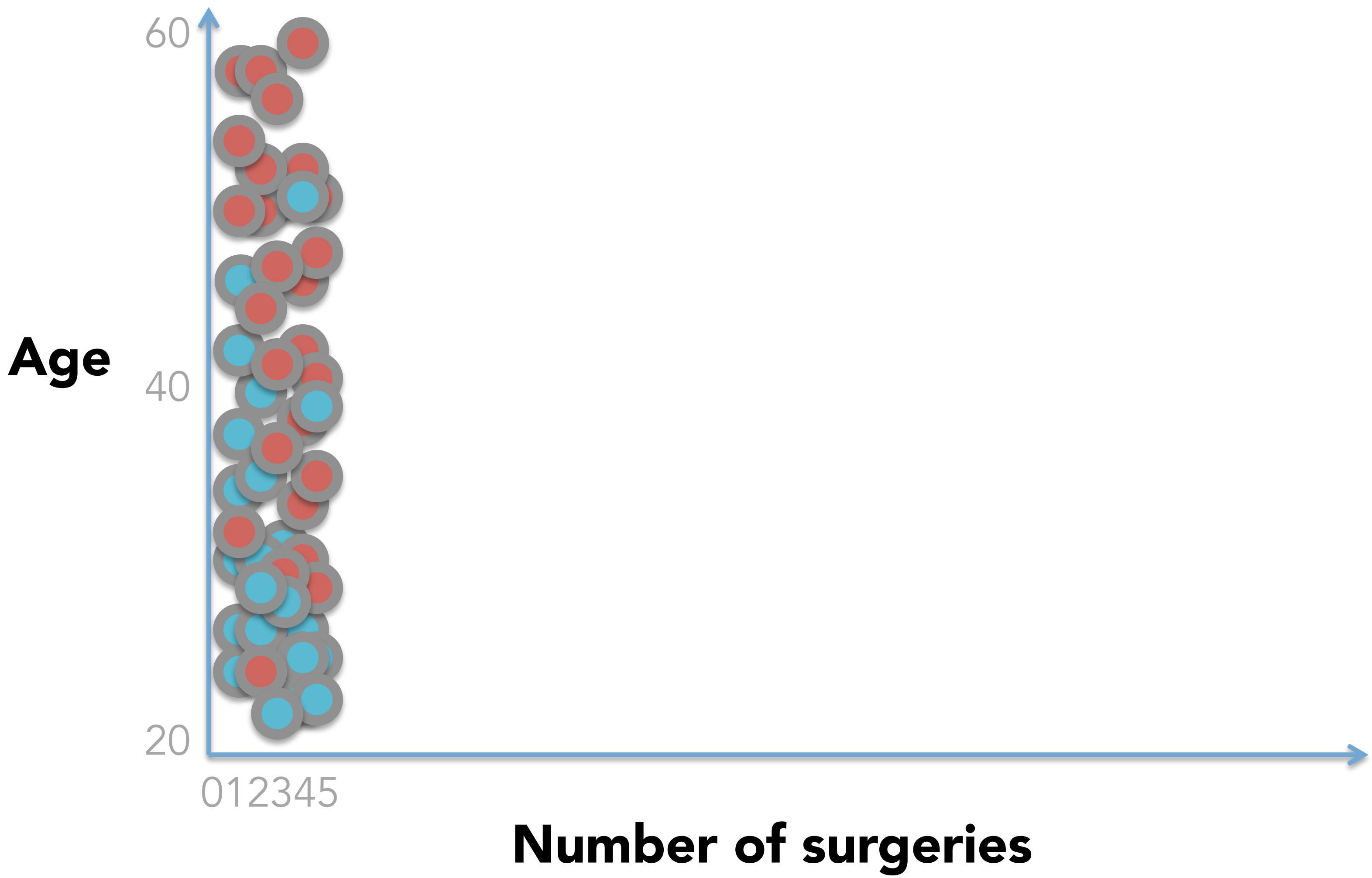
Multiclass KNN Decision Boundary

K=5

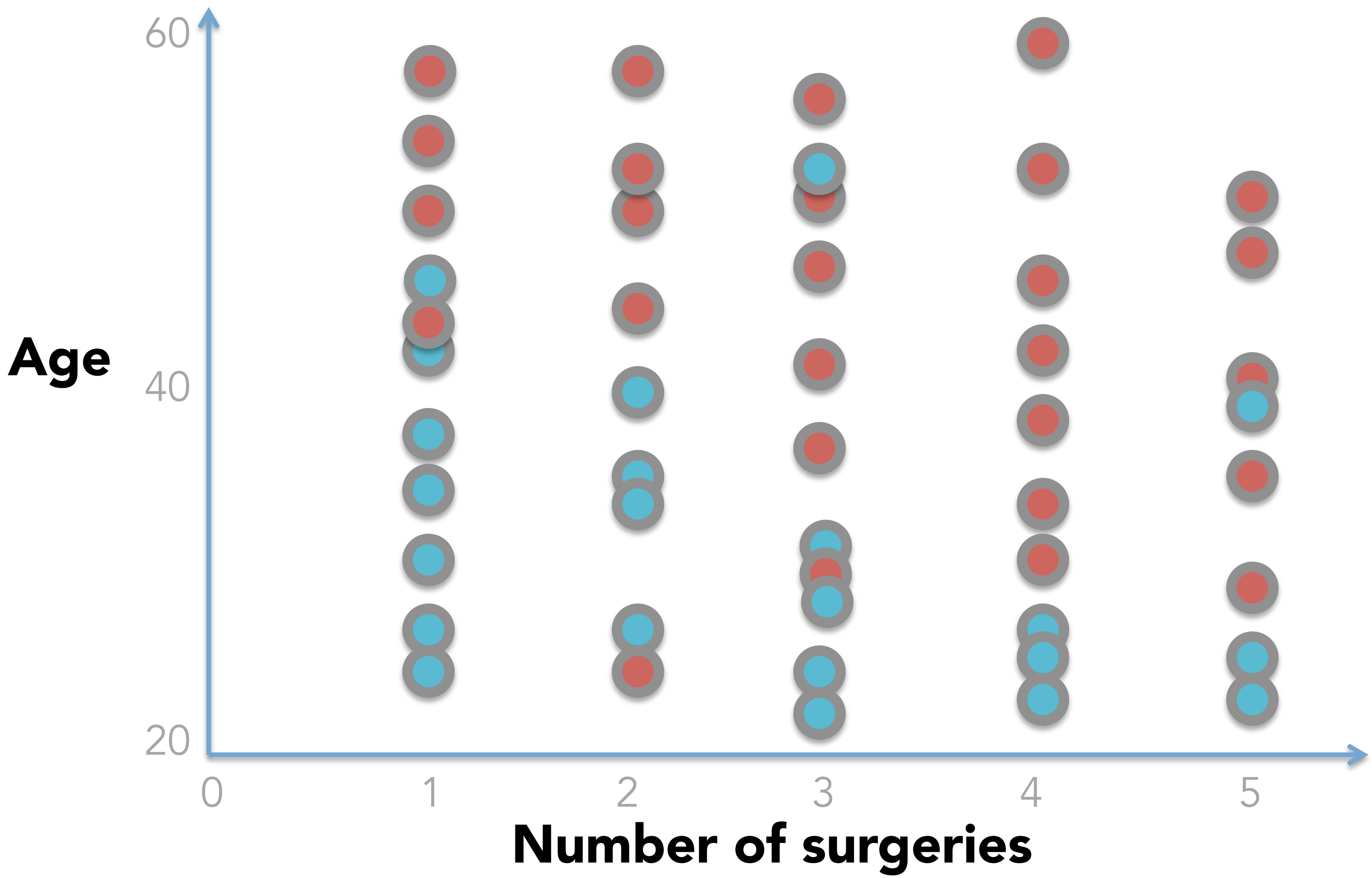


```python
from sklearn.neighbors import KNeighborsClassifier

# Interface not different from LinearRegression at all
model = KNeighborsClassifier(n_neighbors=5)
model.fit(X_train, Y_train)
Y_pred = model.predict(X_test)
```

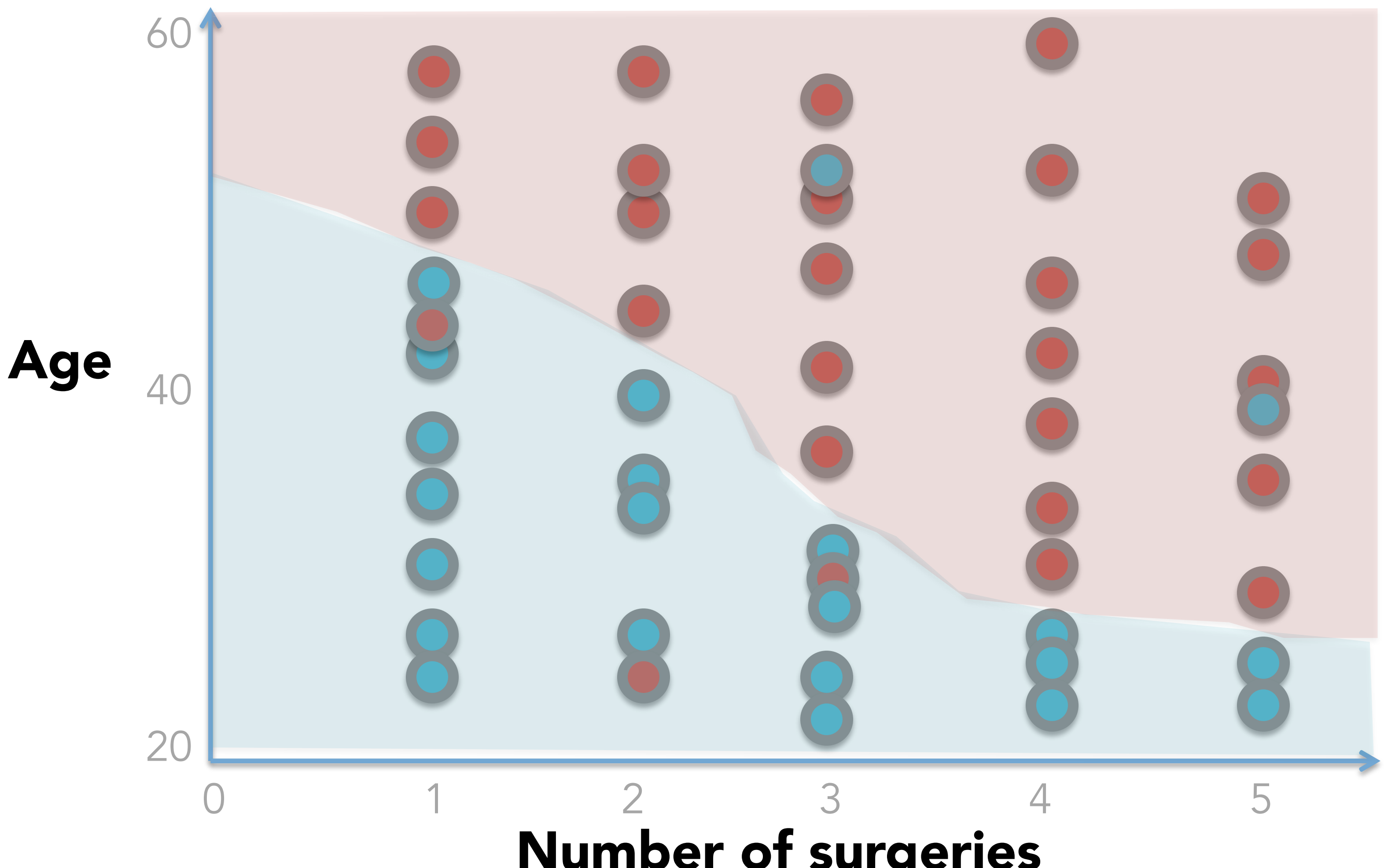
Scaling is crucial!



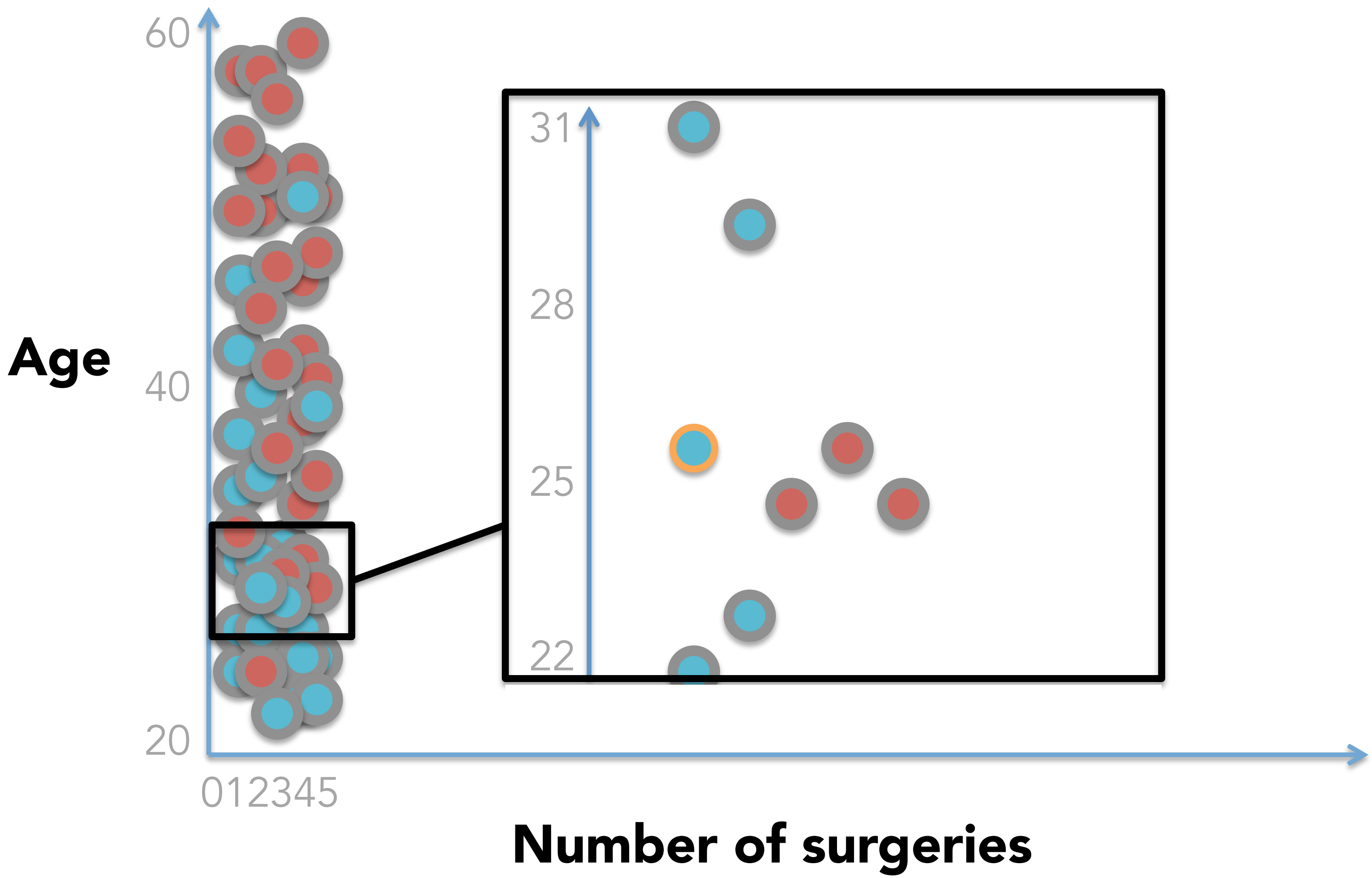
Scaling is crucial!



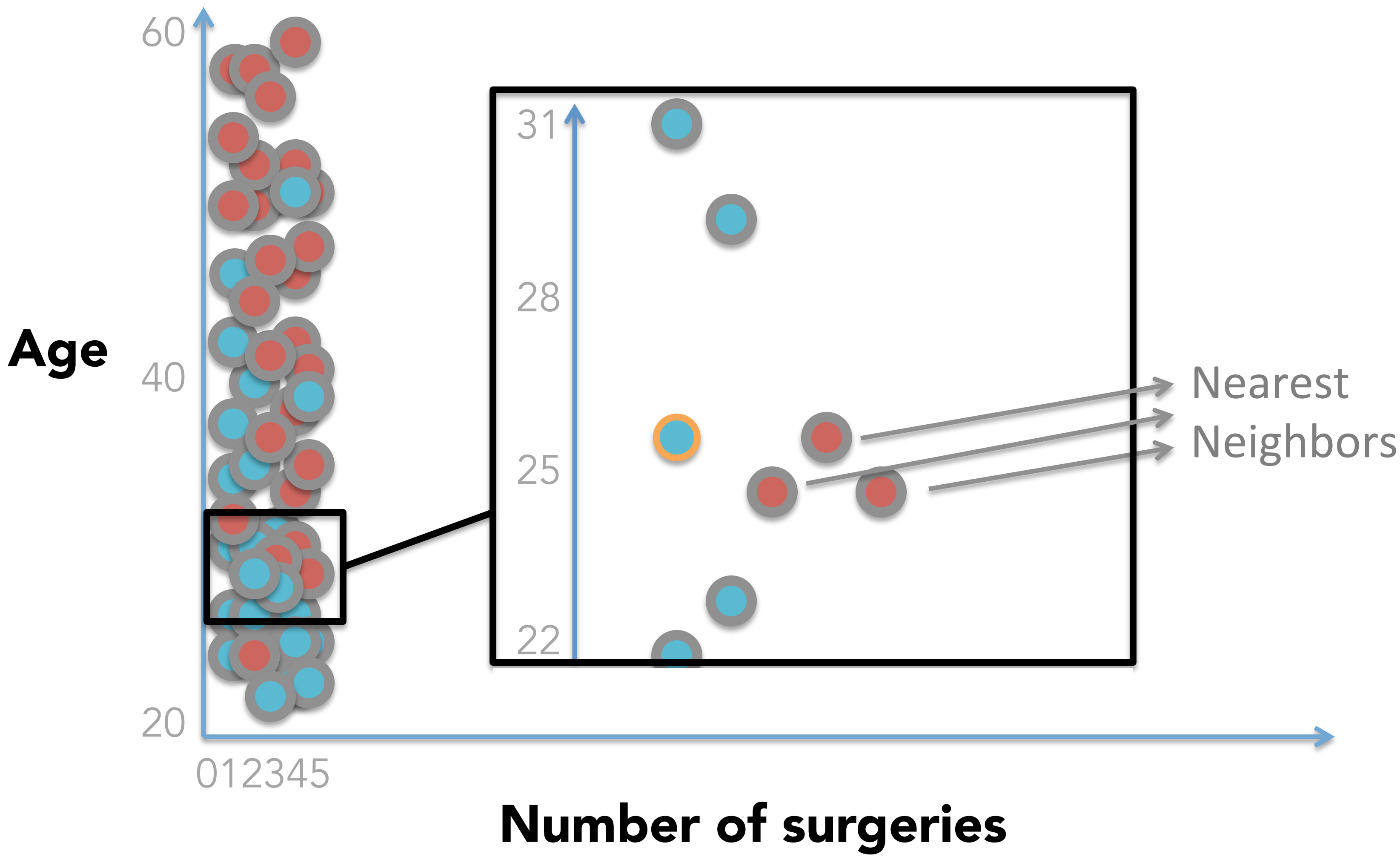
Scaling is crucial!



Scaling is crucial!



Scaling is crucial!



```python
from sklearn.preprocessing import scale

X = scale( X )
```

Logistic Regression

eager

fits slow, predicts fast

low memory (saves only the β values)

K Nearest Neighbors

Lazy

"fits" fast, predicts slow

higher memory (saves entire training set)

scikit.learn.

algorithms galore.