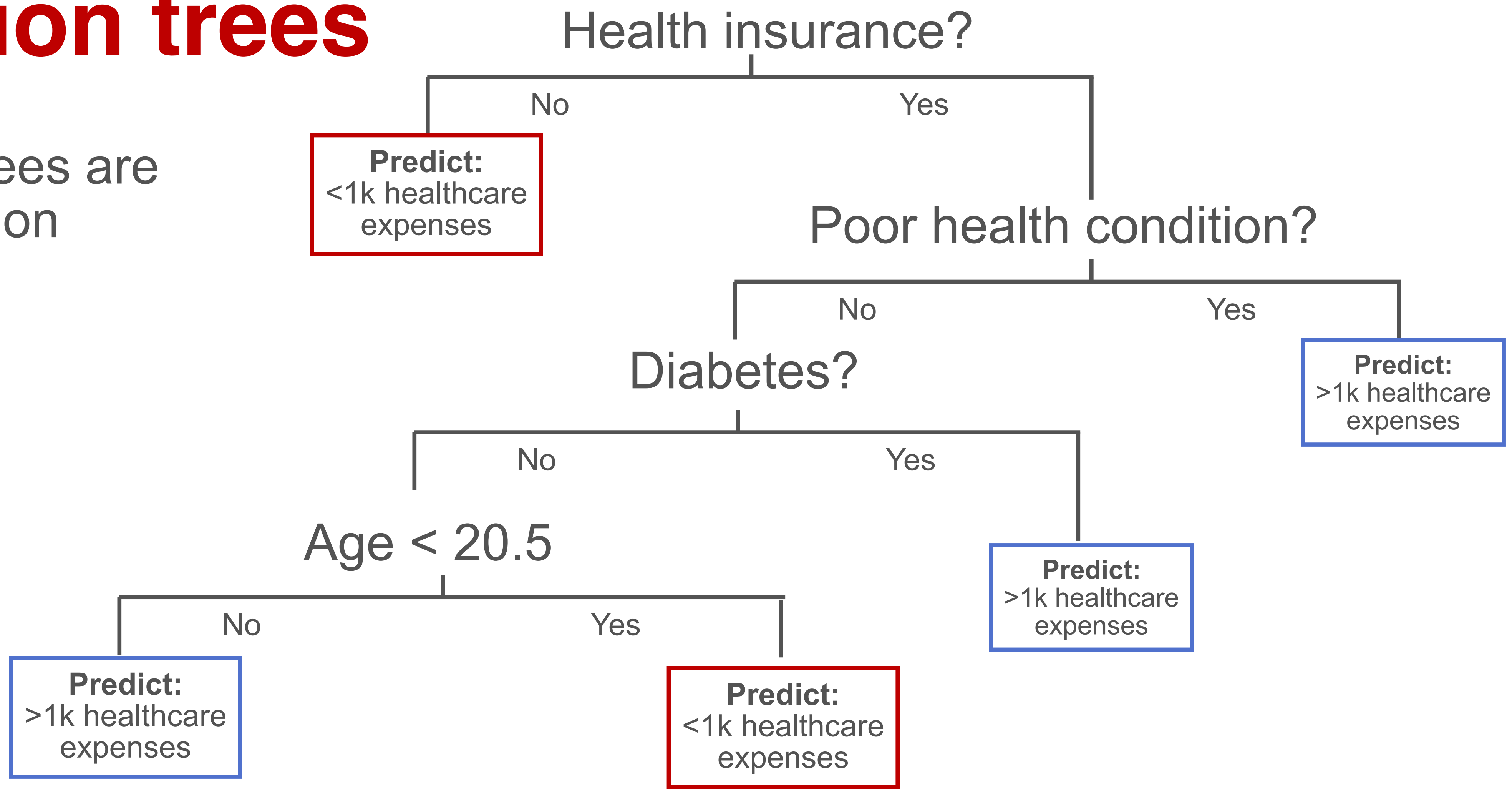
Decision Trees

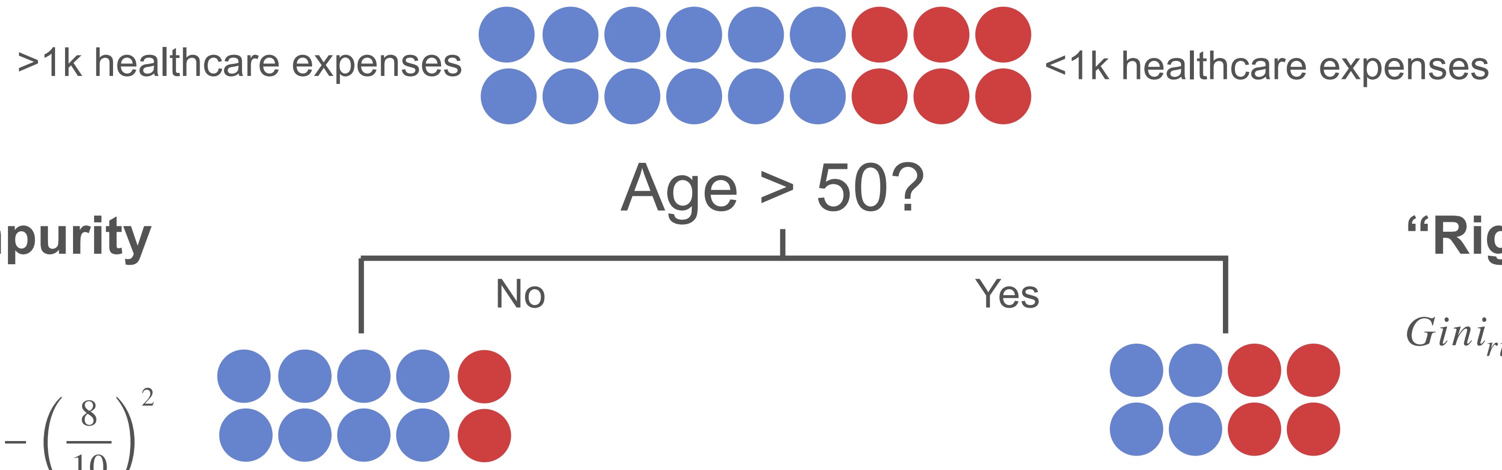
Dr Rebecca Barter

Decision trees

Decision trees are like prediction flowcharts



Training a decision tree with the CART algorithm
(CART = Classification and Regression Tree)



"Left" Gini impurity

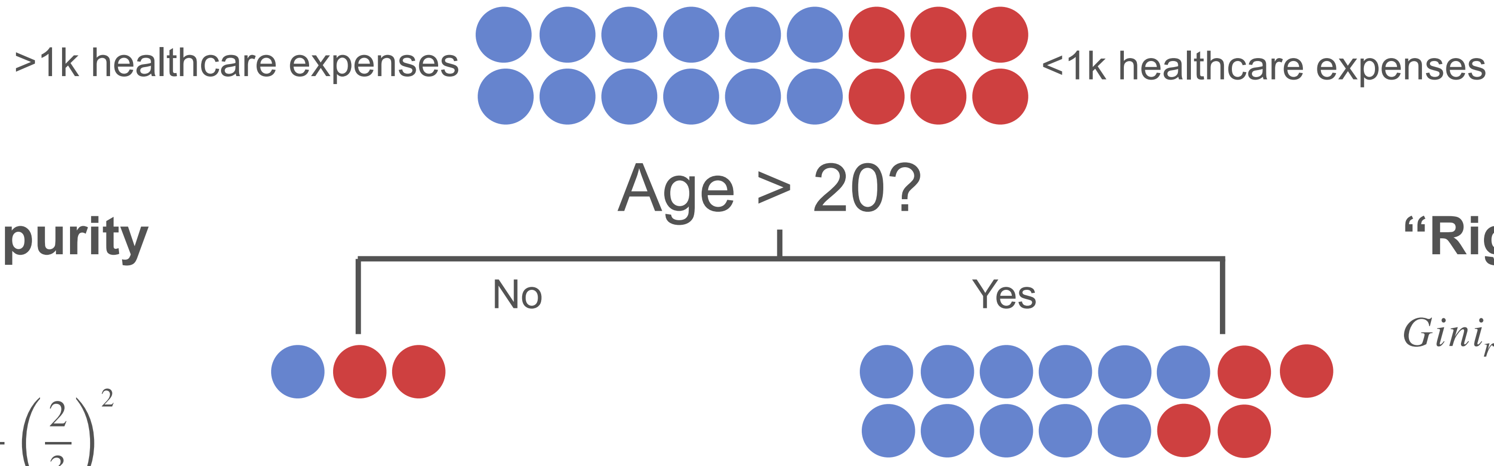
$$Gini_{left} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{2}{10}\right)^2 - \left(\frac{8}{10}\right)^2$$
$$= 0.32$$

"Right" Gini impurity

$$Gini_{right} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{4}{8}\right)^2 - \left(\frac{4}{8}\right)^2$$
$$= 0.5$$

"Total split" Gini impurity $\quad \frac{n_{left}}{n_{total}} Gini_{left} + \frac{n_{right}}{n_{total}} Gini_{right} \quad = \frac{10}{18} \times 0.32 + \frac{8}{18} \times 0.5 \quad = 0.4$

Training a decision tree with the CART algorithm
(CART = Classification and Regression Tree)



"Left" Gini impurity

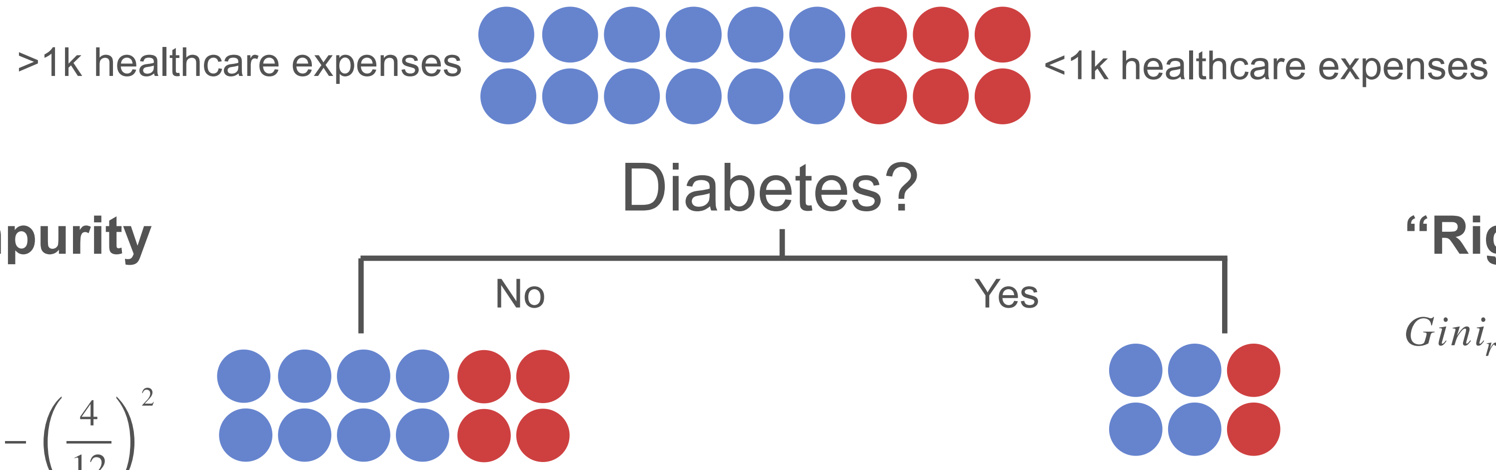
$$Gini_{left} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{1}{3}\right)^2 - \left(\frac{2}{3}\right)^2$$
$$= 0.44$$

"Right" Gini impurity

$$Gini_{right} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{11}{15}\right)^2 - \left(\frac{4}{15}\right)^2$$
$$= 0.39$$

"Total split" Gini impurity $$\frac{n_{left}}{n_{total}} Gini_{left} + \frac{n_{right}}{n_{total}} Gini_{right} = \frac{3}{18} \times 0.44 + \frac{15}{18} \times 0.39 = 0.39$$

Training a decision tree with the CART algorithm
(CART = Classification and Regression Tree)



"Left" Gini impurity

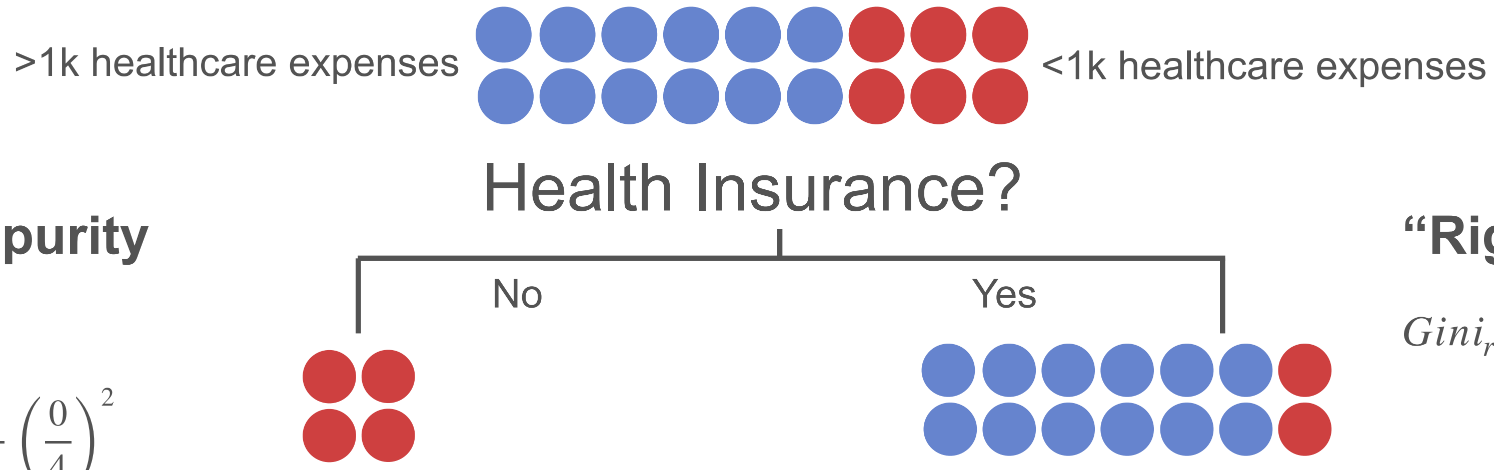
$$Gini_{left} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{8}{12}\right)^2 - \left(\frac{4}{12}\right)^2$$
$$= 0.44$$

"Right" Gini impurity

$$Gini_{right} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{4}{6}\right)^2 - \left(\frac{2}{6}\right)^2$$
$$= 0.44$$

"Total split" Gini impurity $$\frac{n_{left}}{n_{total}} Gini_{left} + \frac{n_{right}}{n_{total}} Gini_{right} = \frac{3}{18} \times 0.44 + \frac{15}{18} \times 0.39 = 0.44$$

Training a decision tree with the CART algorithm
(CART = Classification and Regression Tree)



"Left" Gini impurity

$$Gini_{left} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{4}{4}\right)^2 - \left(\frac{0}{4}\right)^2$$
$$= 0$$

"Right" Gini impurity

$$Gini_{right} = 1 - p_0^2 - p_1^2$$
$$= 1 - \left(\frac{12}{14}\right)^2 - \left(\frac{2}{14}\right)^2$$
$$= 0.24$$

"Total split" Gini impurity $\frac{n_{left}}{n_{total}} Gini_{left} + \frac{n_{right}}{n_{total}} Gini_{right} \quad = \frac{4}{18} \times 0 + \frac{14}{18} \times 0.24 \quad = 0.19$

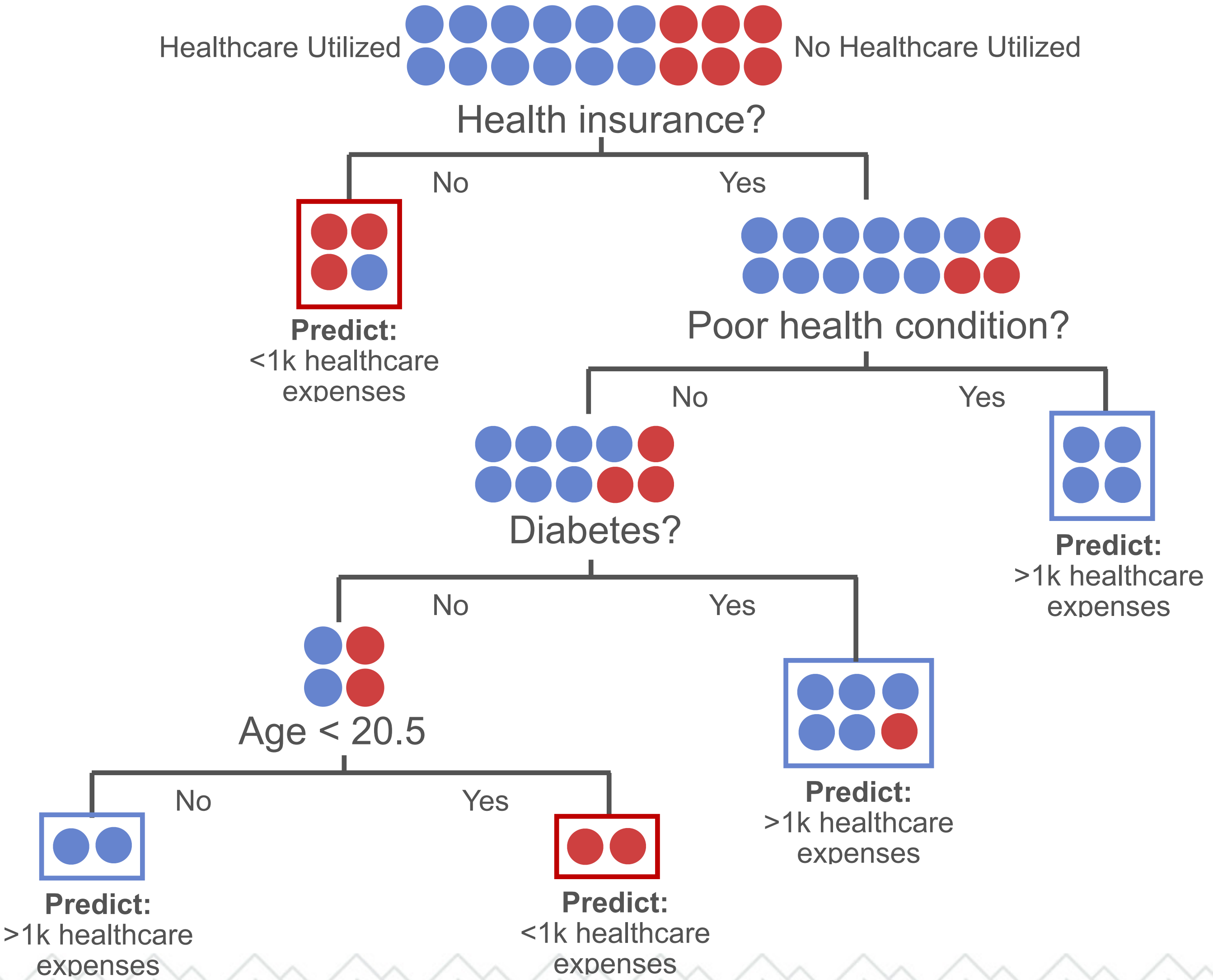
Training a decision tree with the CART algorithm

Options for the first split	Gini impurity
Age > 50? No / Yes	0.4
Age > 20? No / Yes	0.39
Diabetes? No / Yes	0.44
Health Insurance? No / Yes	0.19

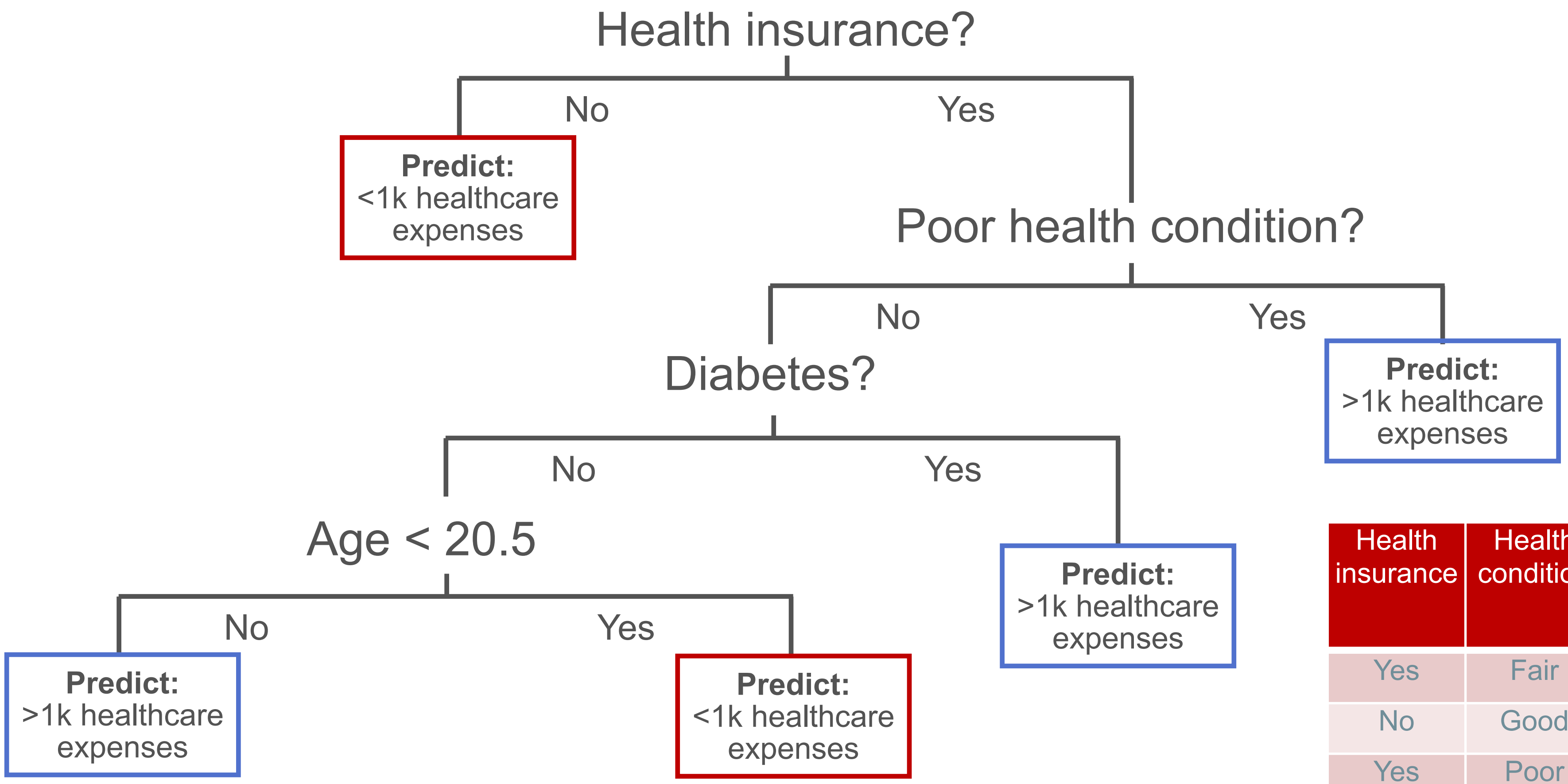
Training a decision tree with the CART algorithm

Options for the first split	Gini impurity
Health Insurance? No / Yes	0.19

CART Predictions for binary responses



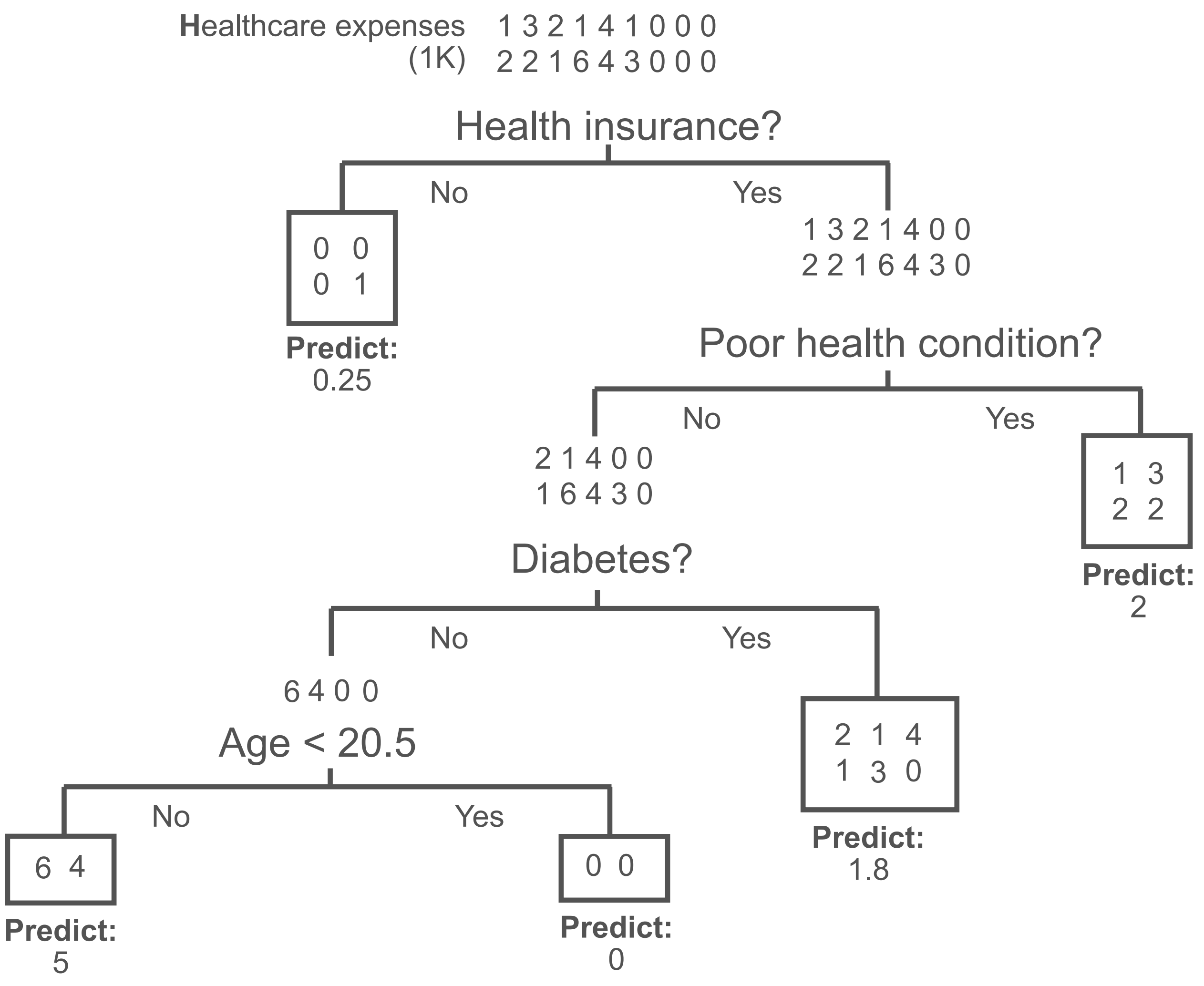
CART Predictions for binary responses



Generate a prediction for a new data point:

Health insurance	Health condition	Diabetes	Age	>1k Healthcare Expenses Prediciton
Yes	Fair	Yes	45	Yes
No	Good	No	22	No
Yes	Poor	Yes	75	Yes
Yes	Excellent	No	54	Yes

CART Predictions for cont. responses



Random Forest

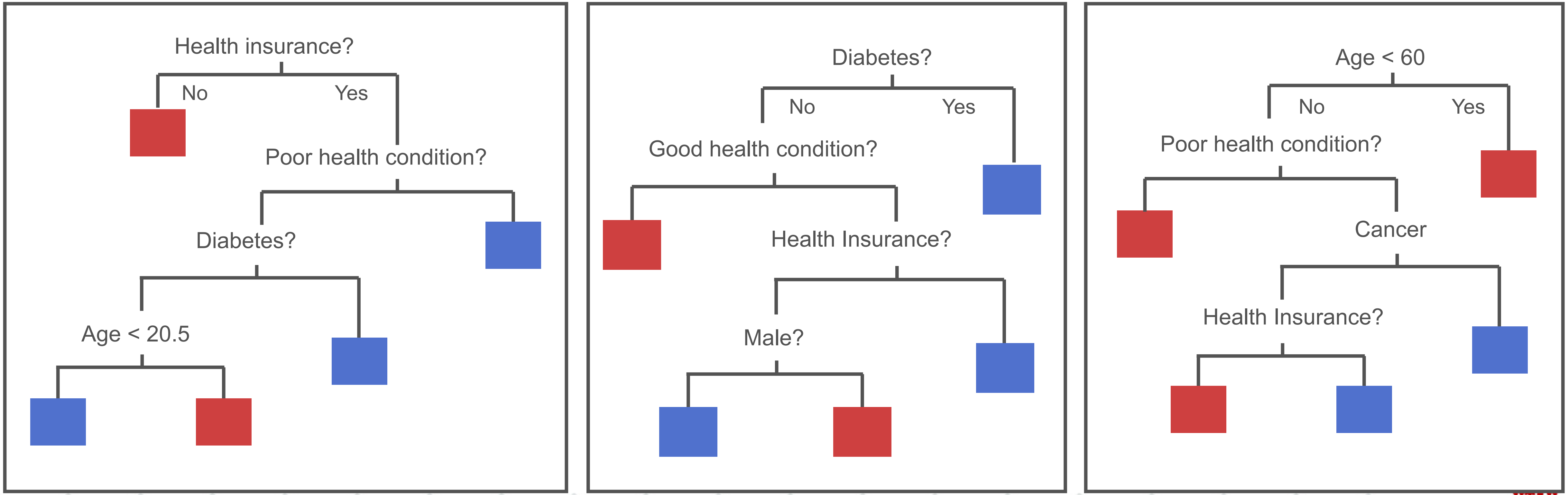
Dr Rebecca Barter

Training a Random Forest (RF) algorithm

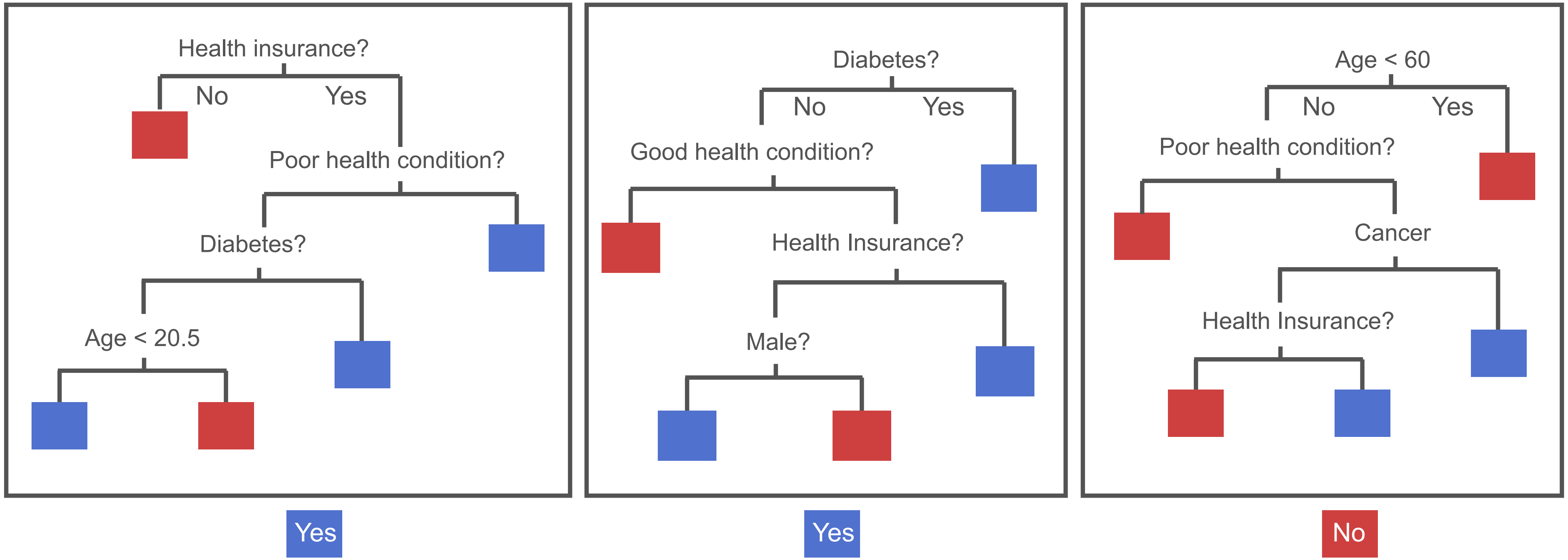
Train multiple different trees “in parallel”/“independently”

(a) Each **tree** is based on a different random sample of the training data

(b) Each **split** considers a different subset of features



Predictions from RF algorithm



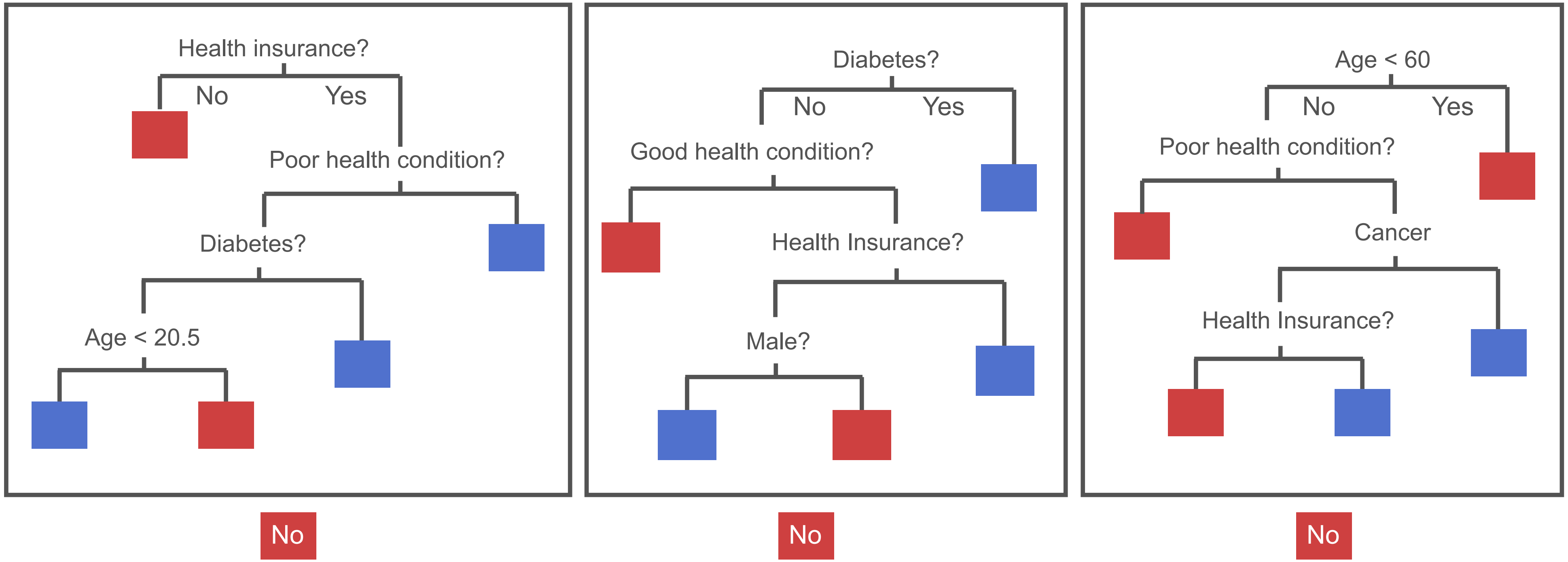
New data point:

Health insurance	Health condition	Diabetes	Age	Gender	Cancer
Yes	Fair	Yes	45	F	N

>1K Healthcare Expenses Prediciton
Yes

Prediction is generated based on “majority vote”

Predictions from RF algorithm



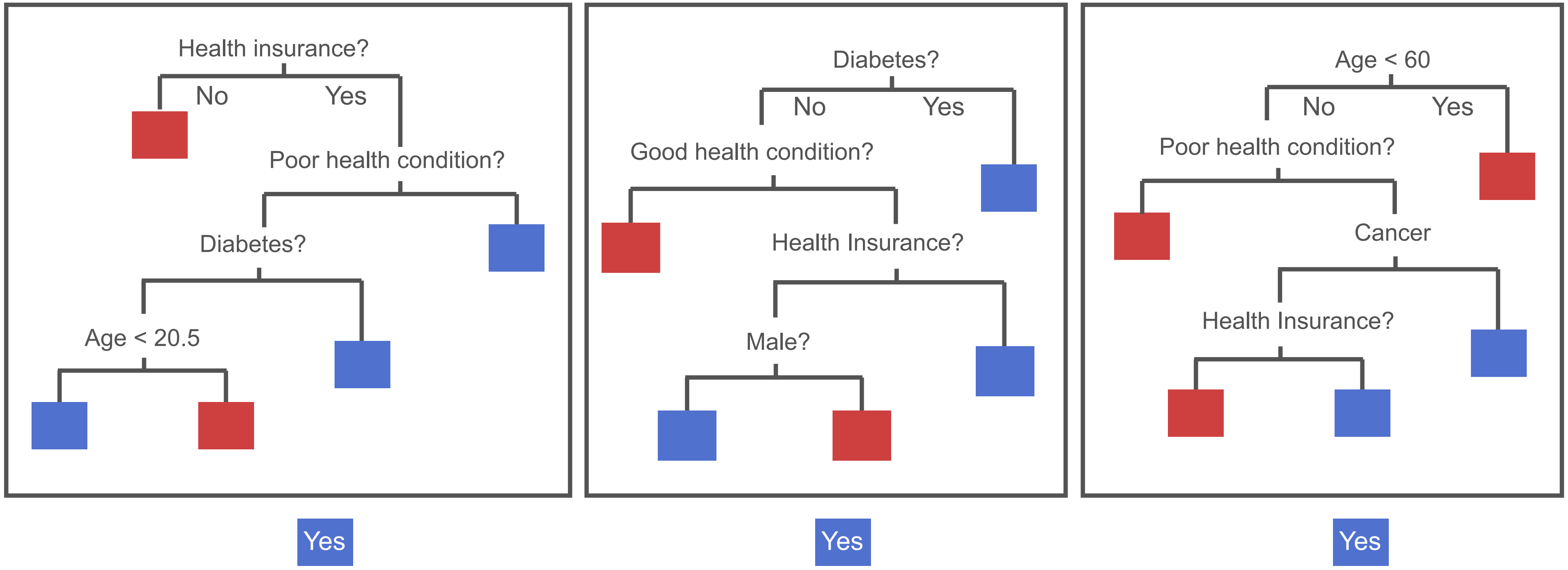
New data point:

Health insurance	Health condition	Diabetes	Age	Gender	Cancer
No	Good	No	22	M	N

>1K Healthcare Expenses Prediciton
No

Prediction is generated based on "majority vote"

Predictions from RF algorithm



New data point:

Health insurance	Health condition	Diabetes	Age	Gender	Cancer
Yes	Poor	Yes	75	M	Y

>1K Healthcare Expenses Prediciton
Yes

Prediction is generated based on "majority vote"

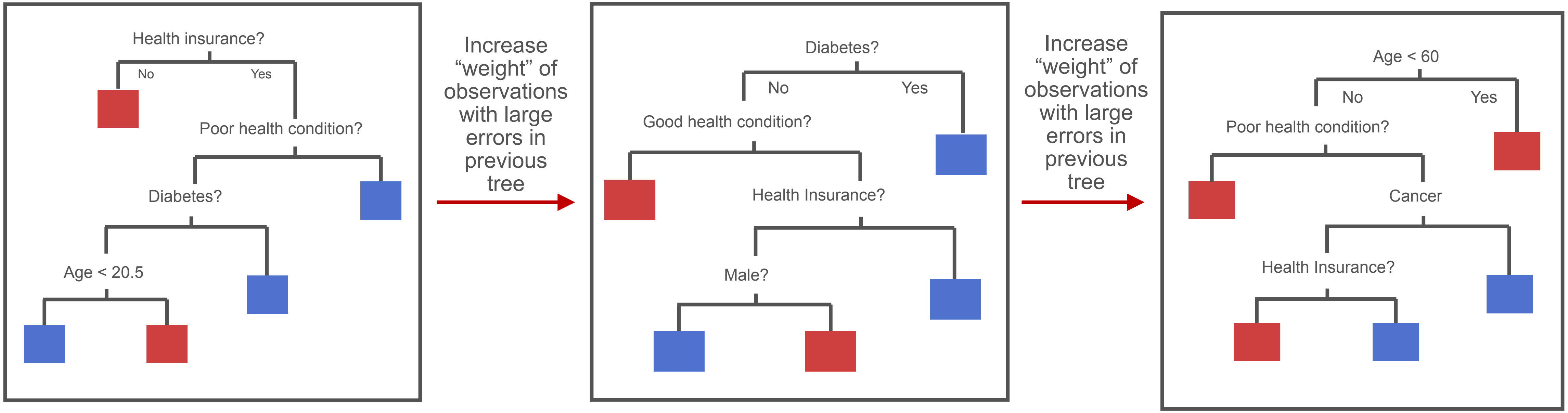
XGBoost
(Extreme Gradient Boosting)

Dr Rebecca Barter

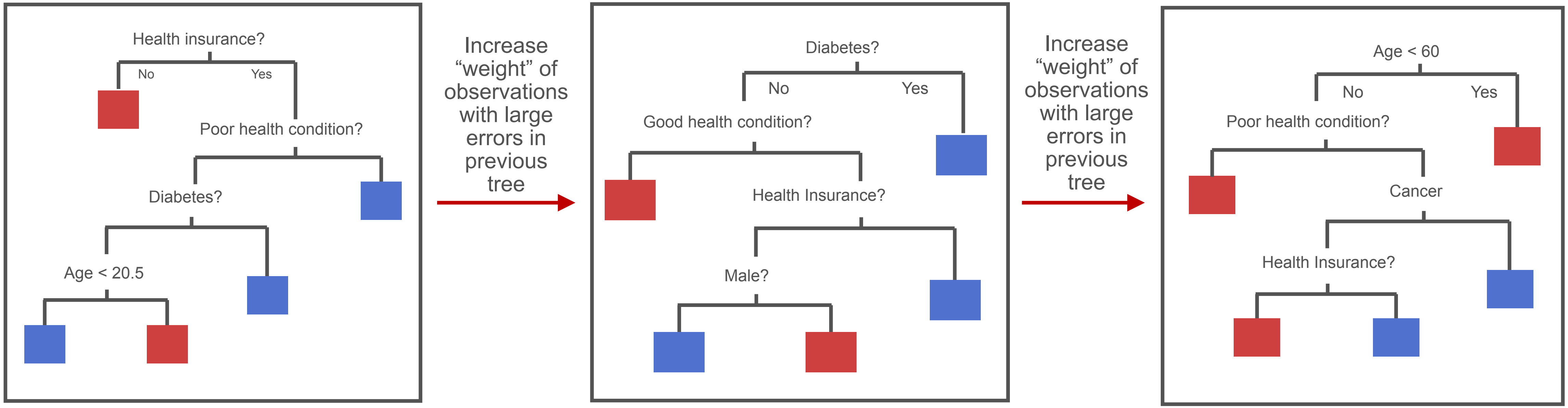
Training an XGBoost algorithm

Train multiple different trees “sequentially”

Each tree is trained to reduce the errors made by the previous trees by placing greater “penalties” on prediction errors for observations with larger prior errors



Predictions from XGBoost algorithm



Weight: 0.1 Yes

Weight: 0.3 Yes

Weight: 0.6 No

Prediction = $0.1 \times 1 \; + \; 0.3 \times 1 \; + \; 0.6 \times 0 \; = \; 0.4 \quad < 0.5$

New data point:

Health insurance	Health condition	Diabetes	Age	Gender	Cancer
No	Good	No	22	M	N

>1K Healthcare Expenses Prediciton
No

Prediction is generated based on <u>weighted</u> “majority vote”

Variable importance

Dr Rebecca Barter

Variable importance for linear models

Variable importance for linear models can be determined from the coefficients, but **coefficients of a linear model are *not* automatically comparable**

Continuous response linear regression

$$\widehat{\text{hlthx}} = -6.59 + 0.07\ \text{age} - 0.13\ \text{sex} + 0.03\ \text{weight} + 1.43\ \text{diabetes}$$

Binary response logistic regression

$$P(\widehat{\text{hlthx1k}}) = \frac{e^{-159.4 + 3.2age - 1.2sex + 0.2weight + 4.8diabetes}}{1 + e^{-159.4 + 3.2age - 1.2sex + 0.2weight + 4.8diabetes}}$$

To determine **feature importance** from a linear model, you must either

(a) **Normalize/standardize each variable** before fitting the linear model

(b) Look at the **theoretical standardized coefficients**

Variable importance for RF and XGBoost

There are two metrics for variable importance for RF and XGBoost models

Permutation importance

How much does the prediction accuracy decrease when you re-train the algorithm after randomly scramble (permute) the values of each variable one at a time?

Gini/Variance ("gain") importance

How much does the Gini impurity (binary) or variance (continuous) decrease across each split involving the variable, averaged over all trees in the forest?