

Introduction to Machine Learning – Decision Trees

Dr. Ab Mosca (they/them)

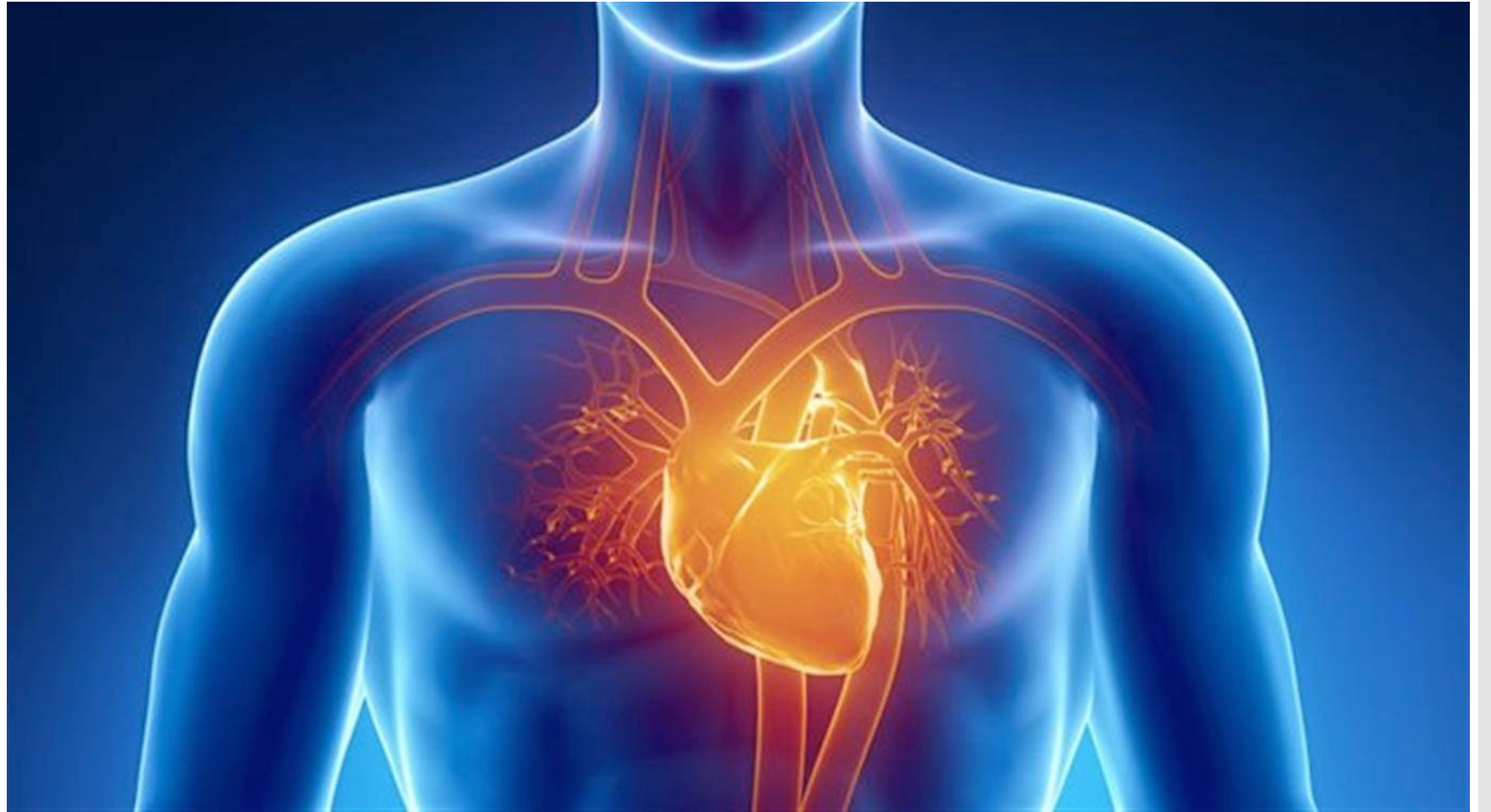
Slides based off slides courtesy of Jordan Crouser (<https://jcrouser.github.io/>)

Plan for Today

Basic mechanics of tree-based methods

- Classification example
- Choosing good splits
- Pruning

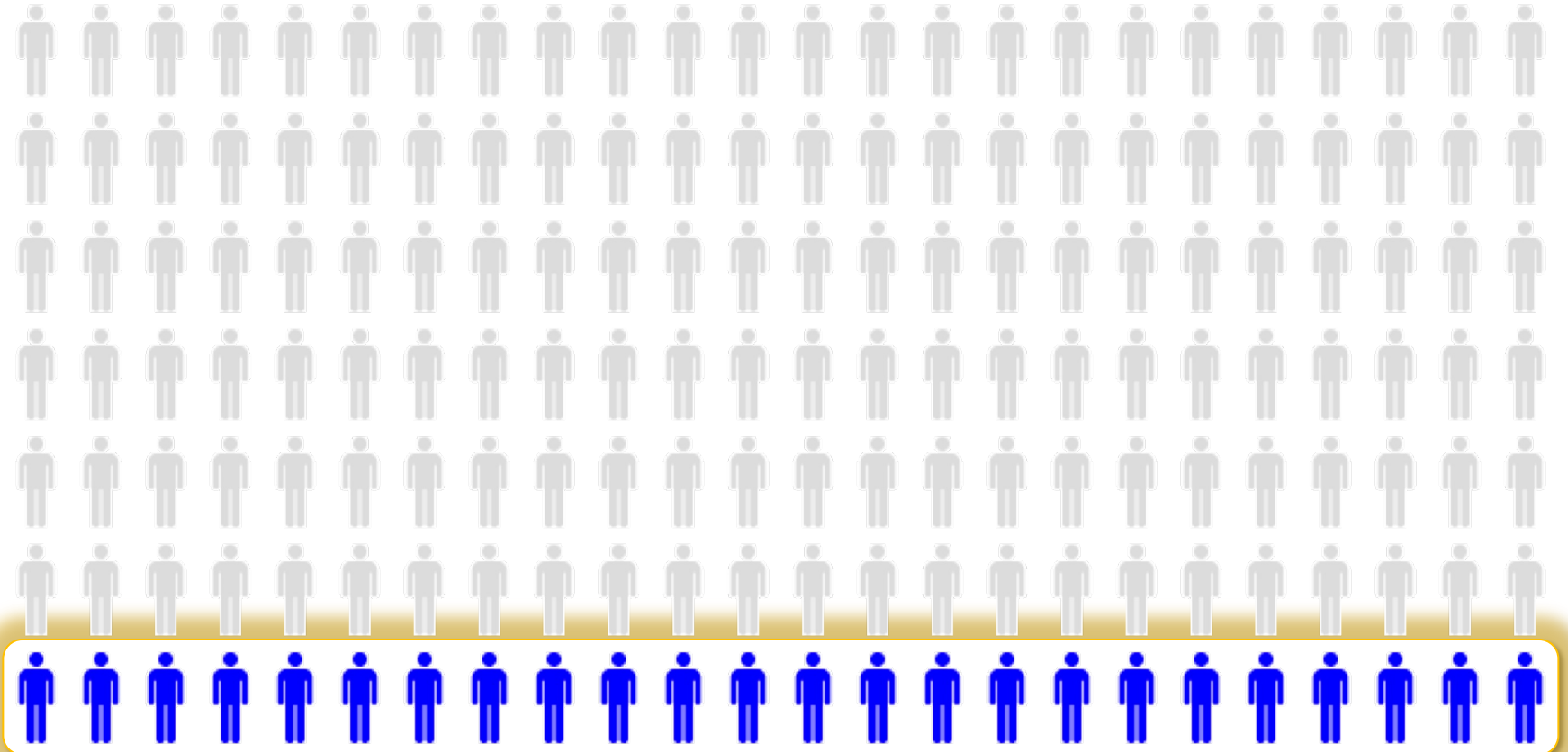
Example:
surviving
cardiac arrest



Example:
surviving
cardiac arrest



Full dataset:
168 patients



24 of 168 patients survived



144 of 168 patients could not be revived

Crystal ball:
best predictor



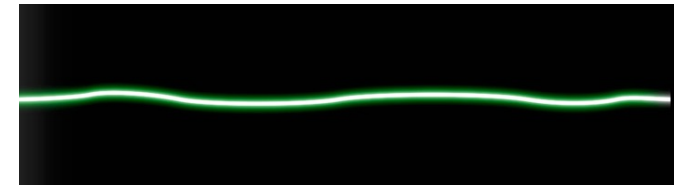
Different types of arrhythmia



Normal

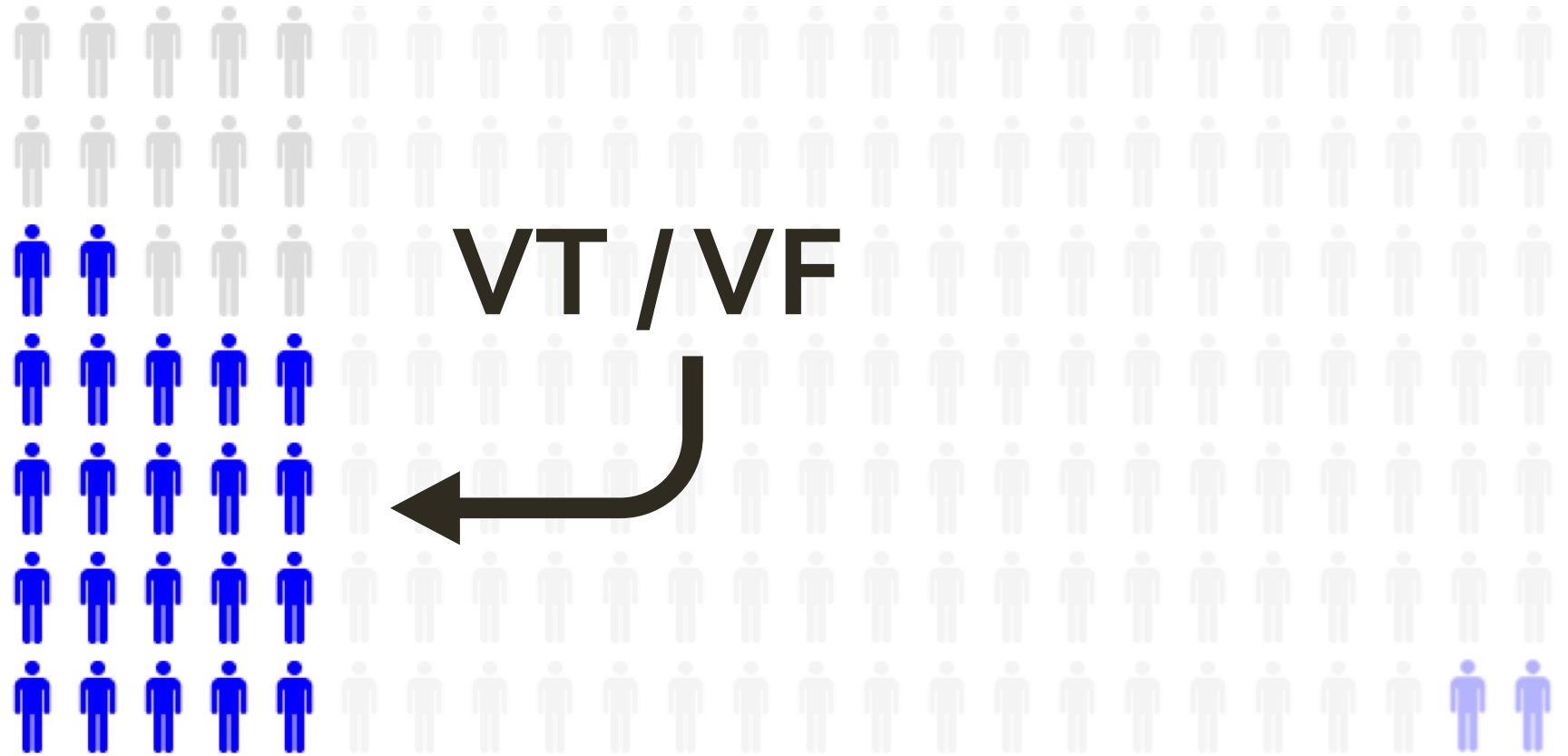


Ventricular Tachycardia (VT) /
Ventricular Fibrillation (VF)



EMD /
Asystole / Other

First Split: Initial Heart Rhythm

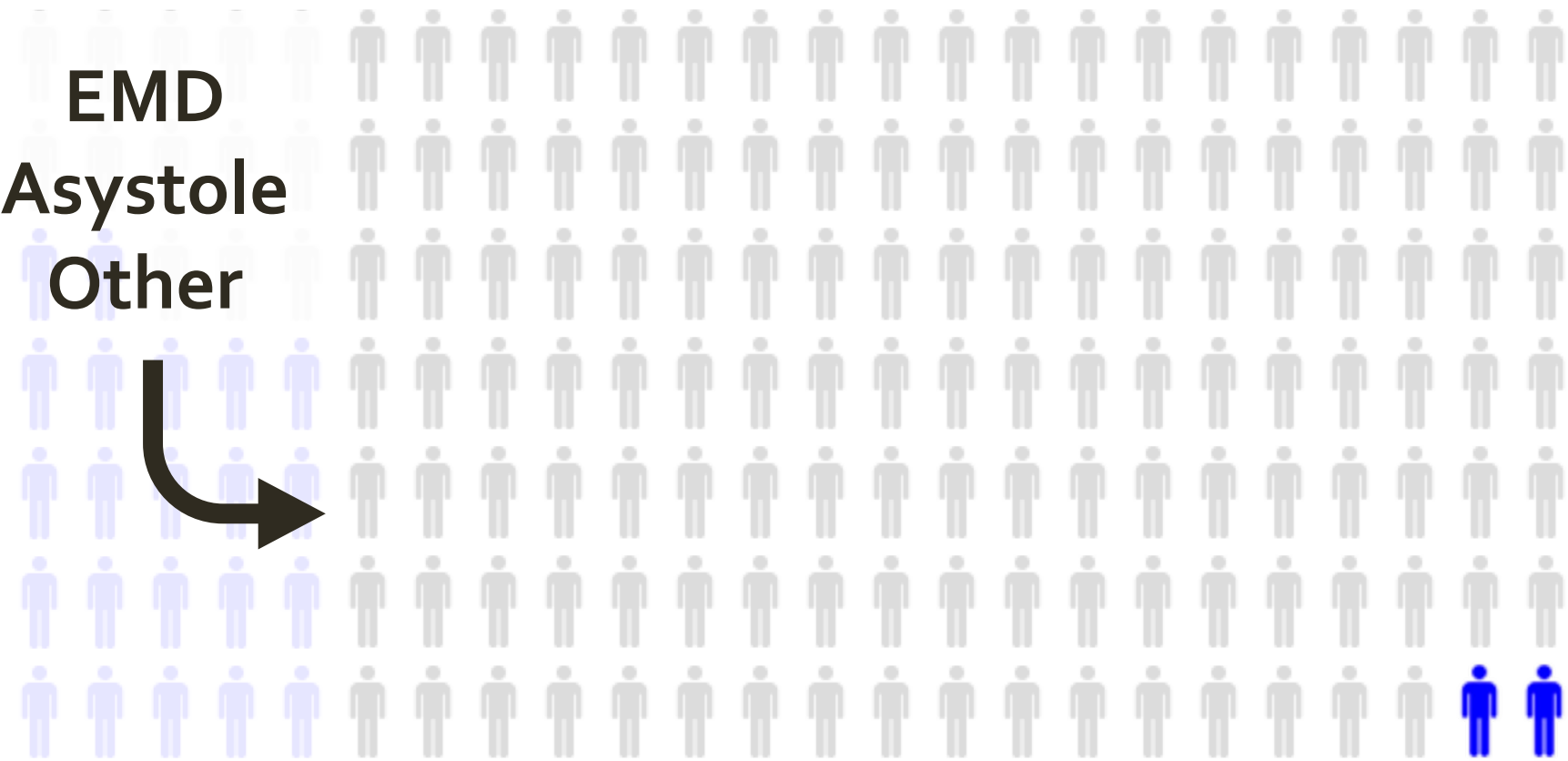


22 of 35 patients survived



13 of 35 patients could not be revived

First Split:
Initial Heart
Rhythm

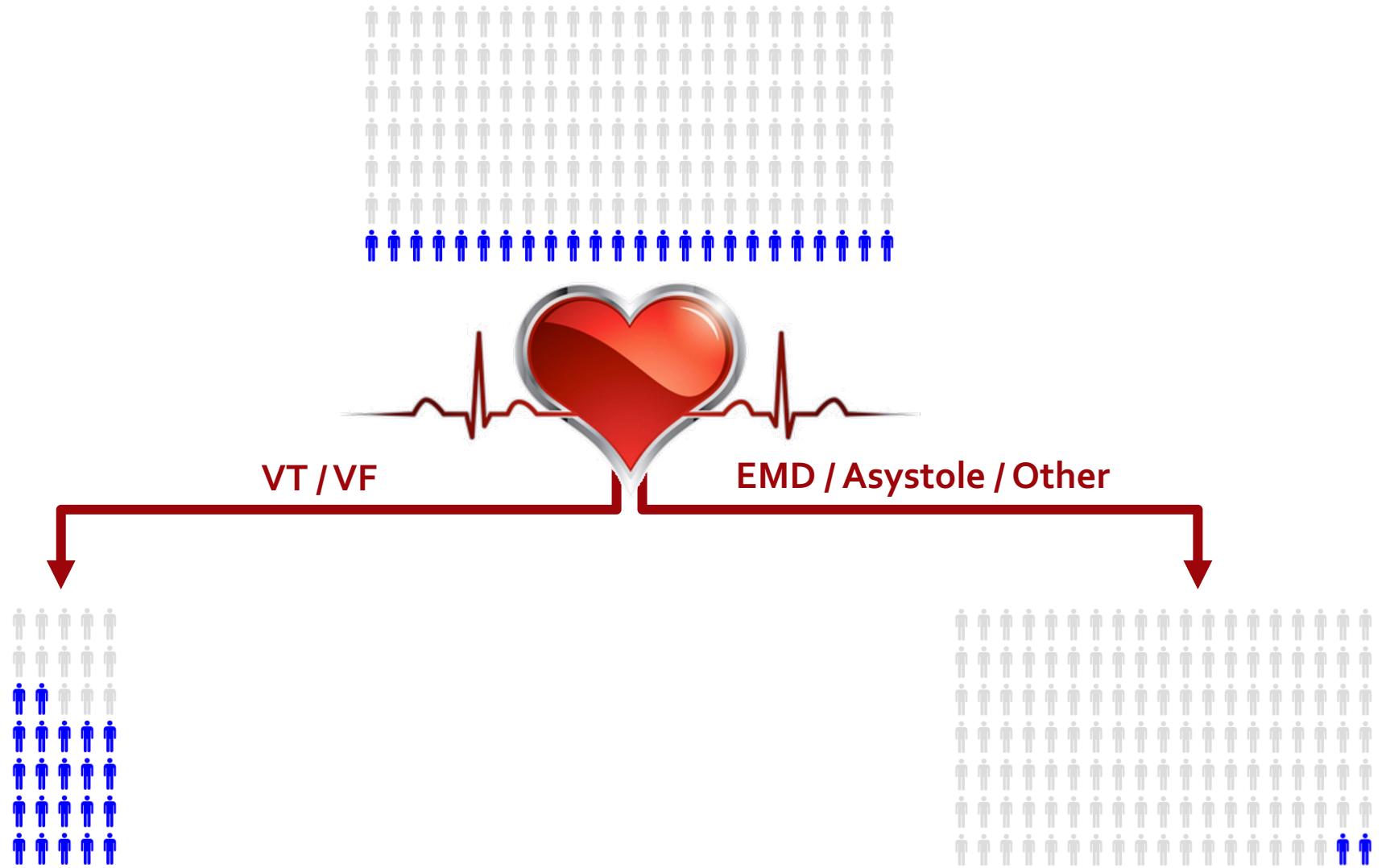


2 of 133 patients survived

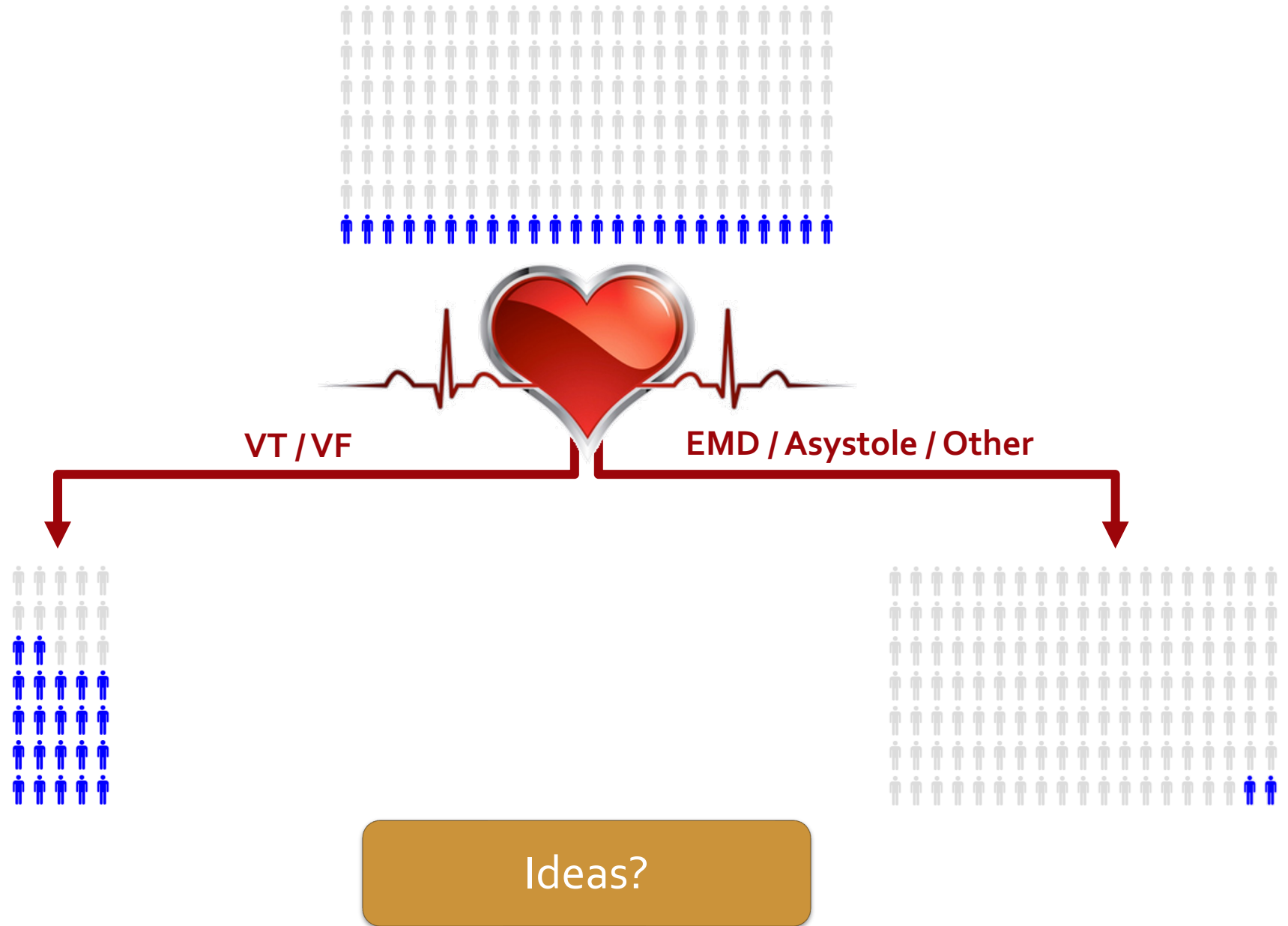


131 of 133 patients could not be revived

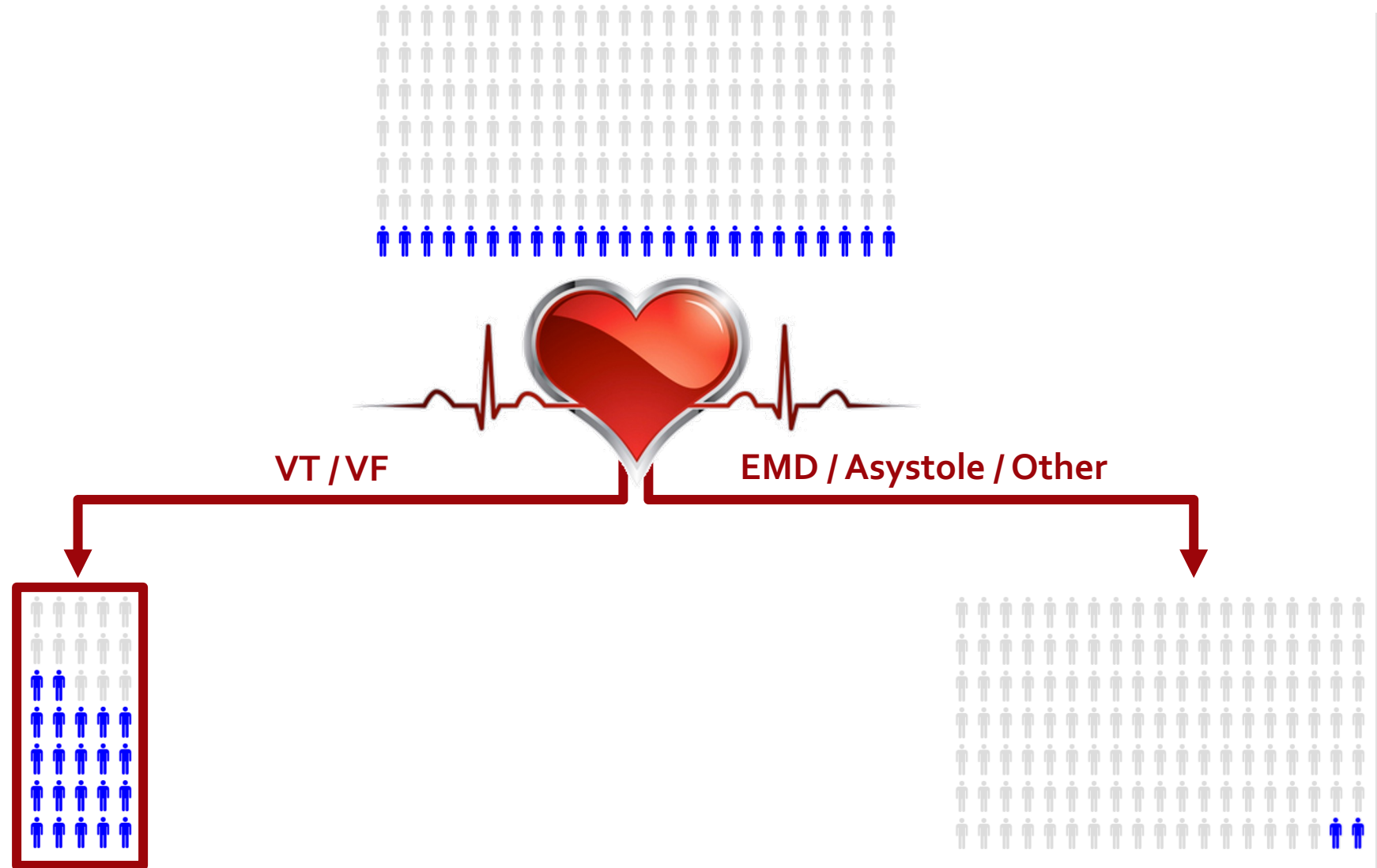
Another view: partitioning



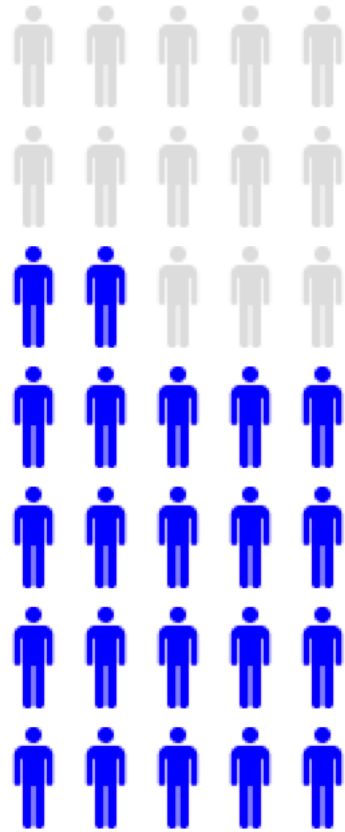
Now what do we do?



Recursion!



VT /VF group
only

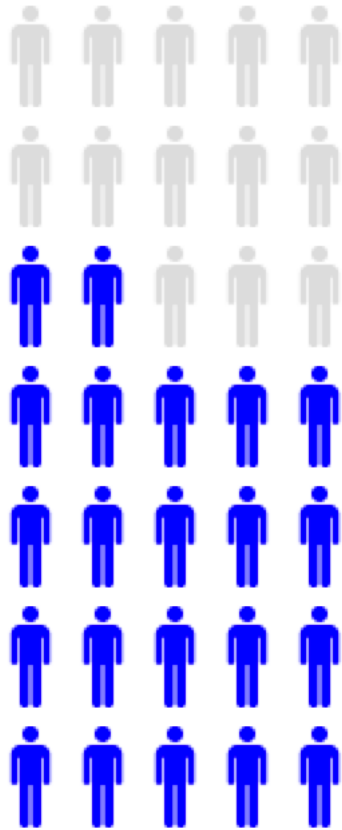


22 of 35 patients survived

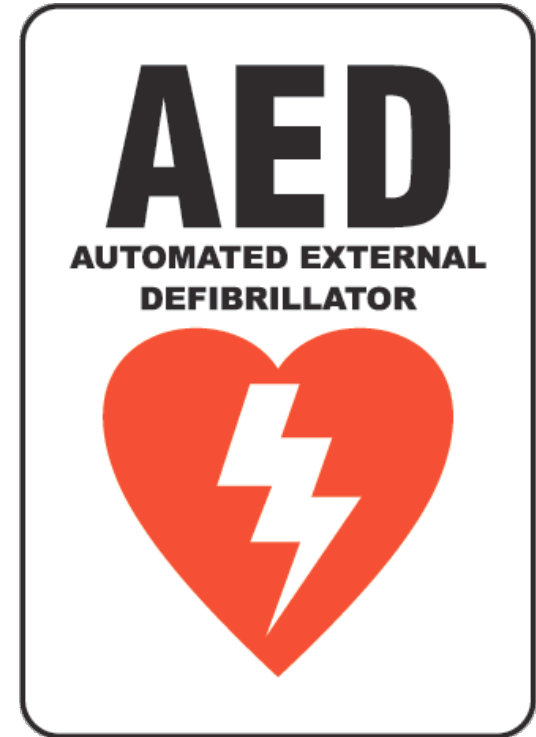


13 of 35 patients could not be revived

Next split:
response to
defibrillation

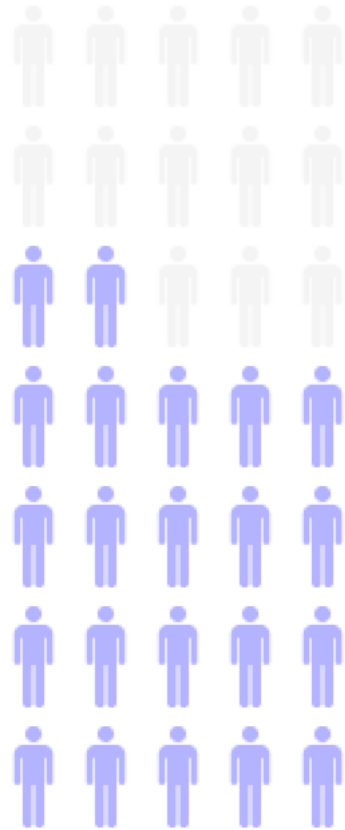


22 of 35 patients survived

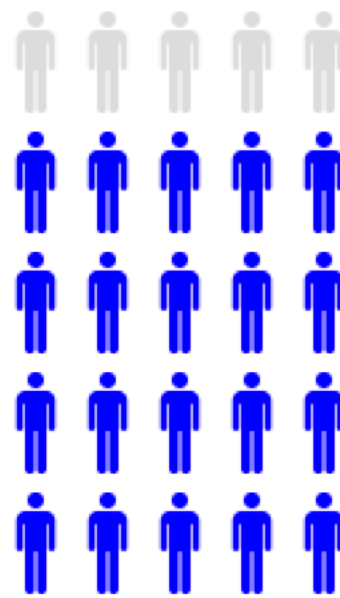


13 of 35 patients could not be revived

Next split:
response to
defibrillation



20 of 25 patients survived

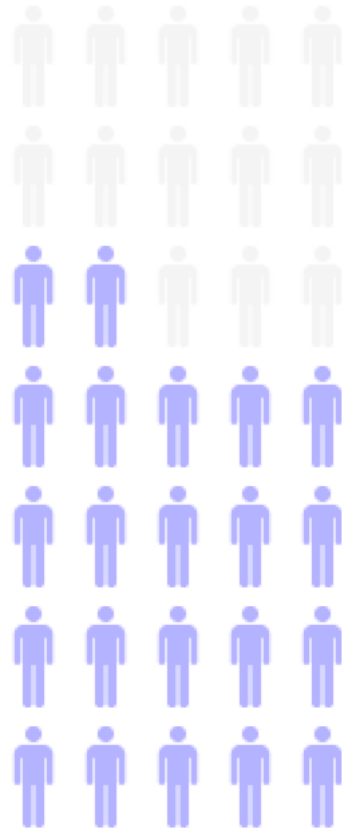


5 of 25 patients could not be revived

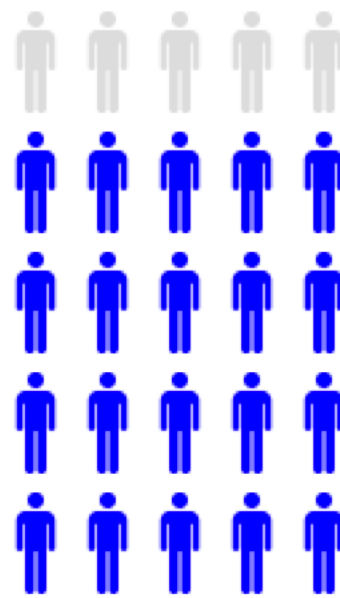
 **Improve**



Next split:
response to
defibrillation

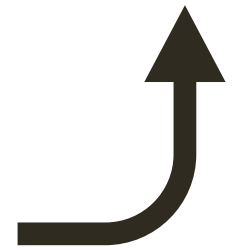


20 of 25 patients survived

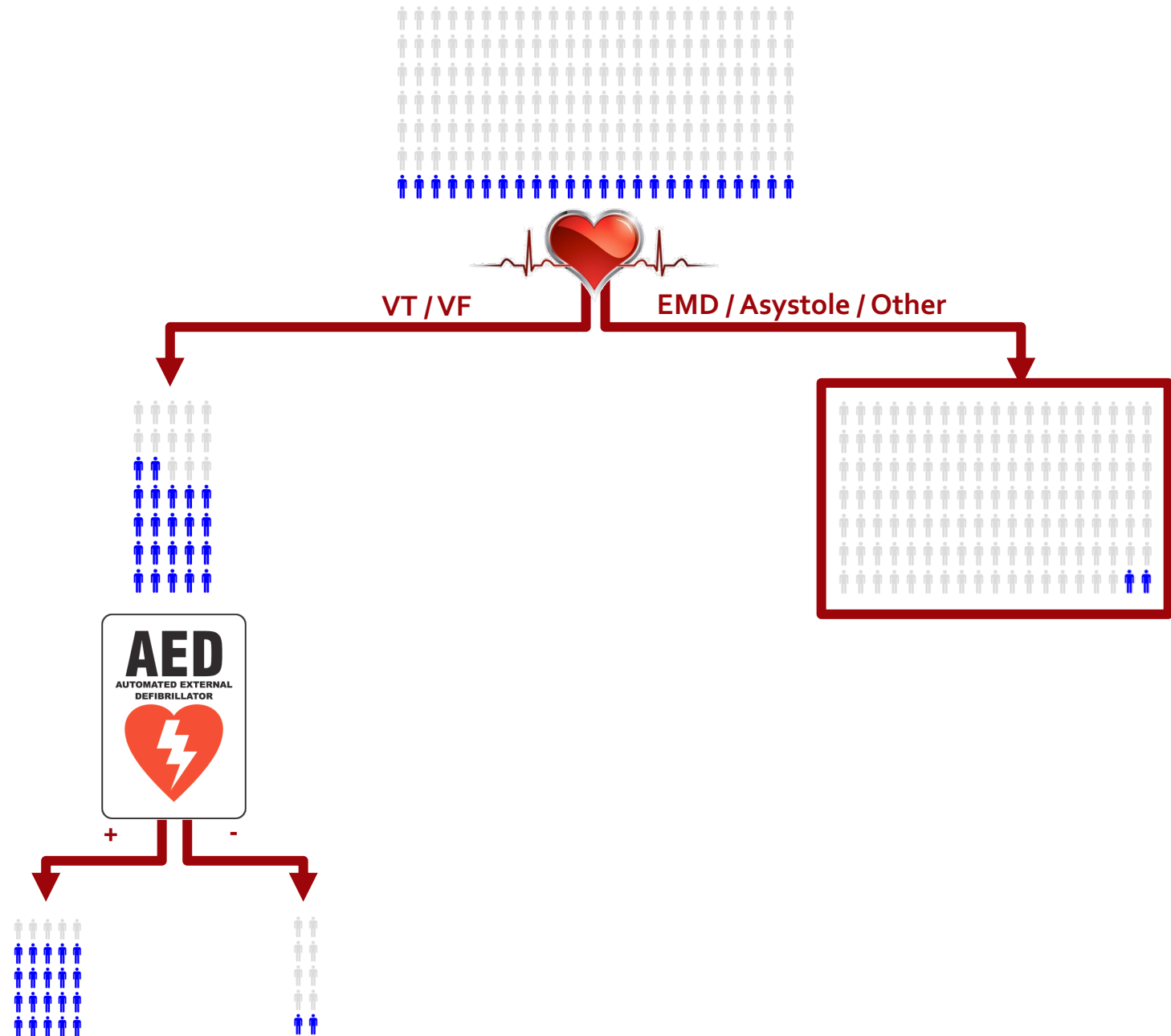


5 of 25 patients could not be revived

Same / Worse



Partition view



Next split: response to ~~defibrillation~~ medication

Next split:
response to
defibrillation



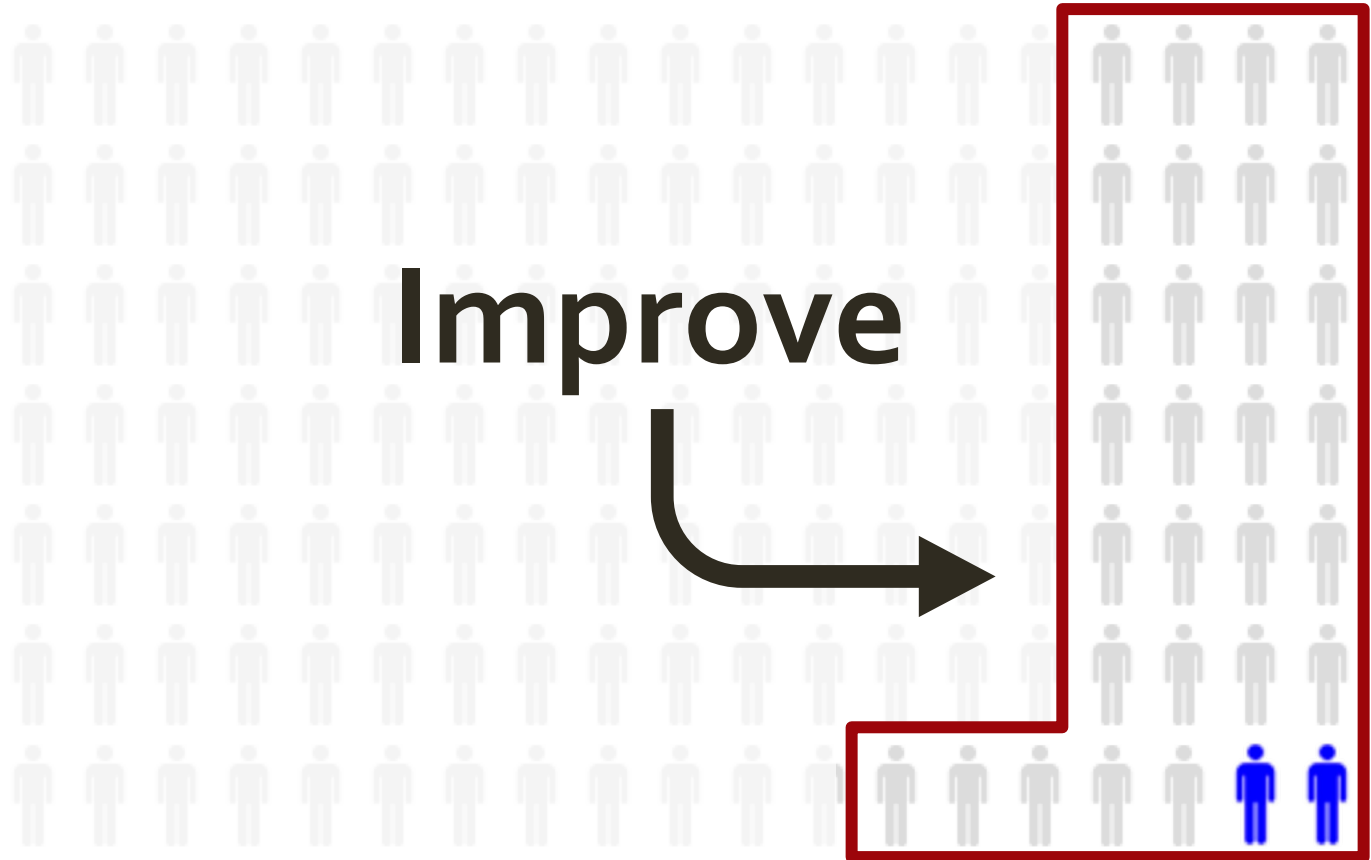
2 of 133 patients survived



131 of 133 patients could not be revived

Next split: response to ~~defibrillation~~ medication

Next split:
response to
defibrillation



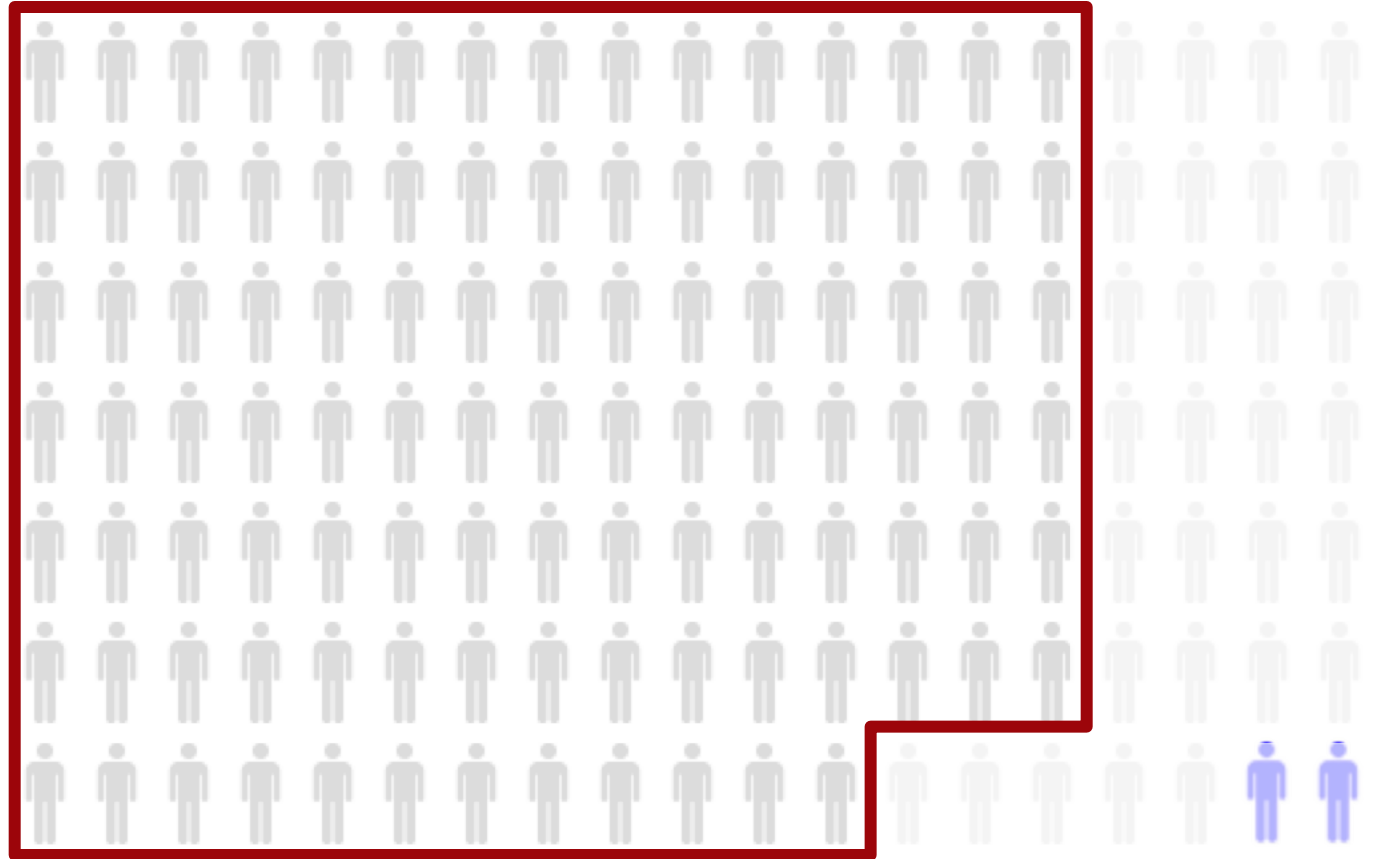
2 of 31 patients survived



29 of 31 patients could not be revived

Next split: response to ~~defibrillation~~ medication

Next split:
response to
defibrillation

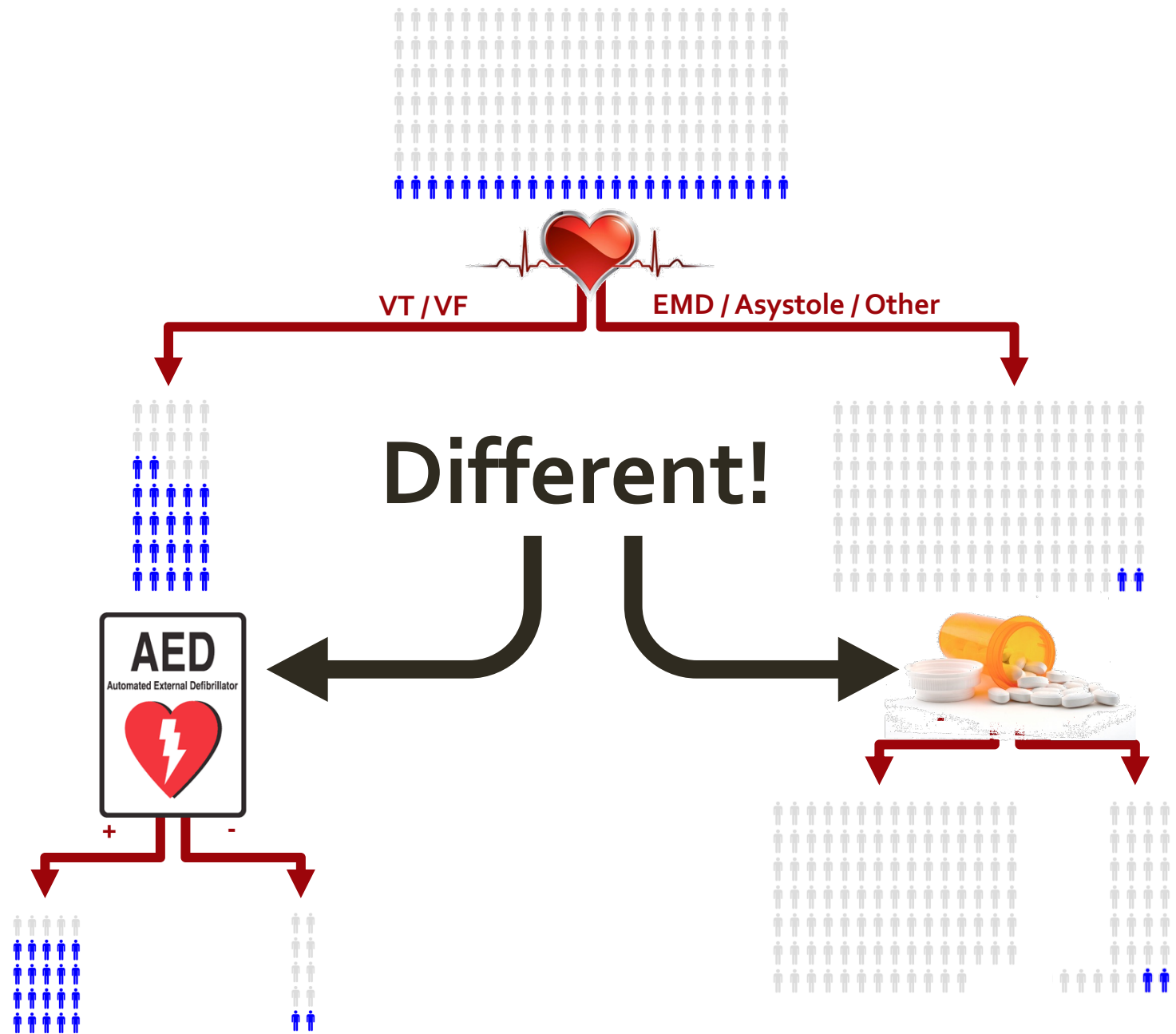


0 of 102 patients survived

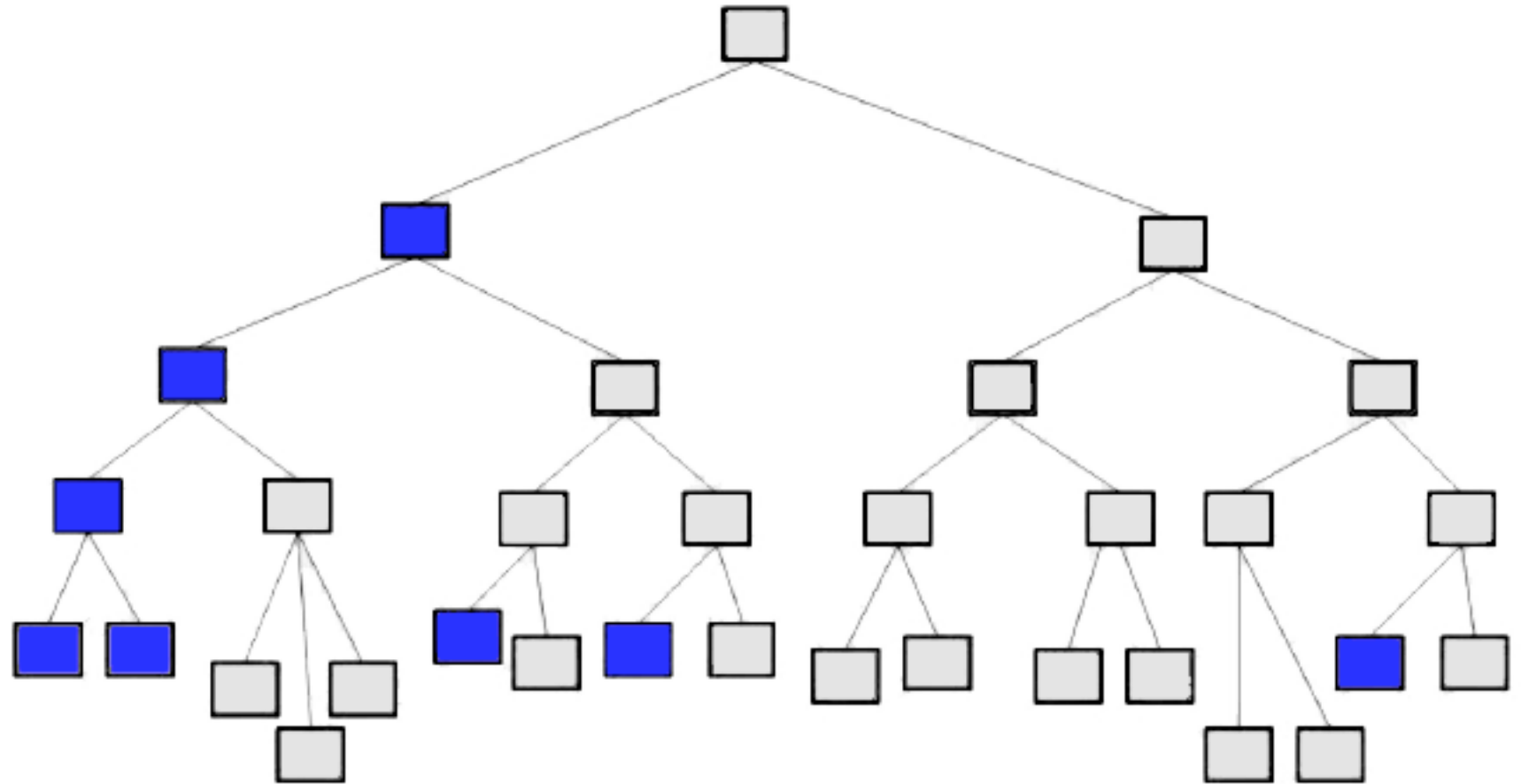


102 of 102 patients could not be revived

Partition view



Continue...



Discussion

What could go wrong with this approach?

Growing (and pruning) trees

Big idea: build a big tree, then cut off (“prune”) the branches that aren’t improving performance

Growing (and pruning) trees

Big idea: build a big tree, then cut off (“prune”) the branches that aren’t improving performance

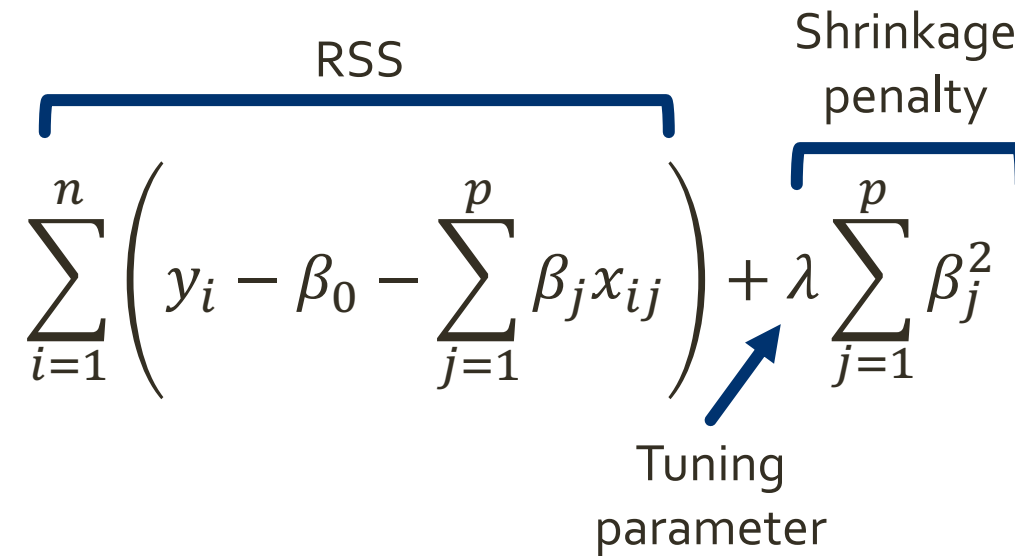
Why not just build a smaller tree to begin with?

Flashback: the lasso

- **Big idea:** minimize RSS plus an additional penalty that rewards small (sum of) **coefficient values**

$$\underbrace{\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2}_{\text{RSS}} + \underbrace{\lambda \sum_{j=1}^p \beta_j^2}_{\text{Shrinkage penalty}}$$

Tuning parameter

The diagram shows the Lasso regression equation. A blue bracket above the first term, $\sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2$, is labeled "RSS". A second blue bracket above the second term, $\lambda \sum_{j=1}^p \beta_j^2$, is labeled "Shrinkage penalty". A blue arrow points from the text "Tuning parameter" below to the λ coefficient in the second term.

Cost complexity pruning

- **Big idea:** minimize RSS plus an additional penalty that rewards small **trees**

The diagram illustrates the cost complexity pruning formula, which is a combination of the Residual Sum of Squares (RSS) and a shrinkage penalty. The formula is presented as follows:

$$\sum_{m=1}^{|T|} \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha |T|$$

The formula is annotated with several labels and arrows to explain its components:

- RSS:** A bracket above the first two summations indicates that they together represent the Residual Sum of Squares.
- Shrinkage penalty:** A bracket above the term $\alpha |T|$ identifies it as the shrinkage penalty.
- Loop over all terminal nodes:** An arrow points to the outer summation $\sum_{m=1}^{|T|}$, indicating that the process loops over all terminal nodes of the tree.
- Then consider only the relevant predictors:** An arrow points to the inner summation $\sum_{i: x_i \in R_m}$, indicating that for each terminal node, only the data points belonging to that node are considered.
- Difference b/t true & predicted values:** An arrow points to the squared term $(y_i - \hat{y}_{R_m})^2$, representing the squared difference between the true value y_i and the predicted value \hat{y}_{R_m} .
- Rewards trees with fewer terminal nodes:** An arrow points to the term $\alpha |T|$, explaining that this term rewards trees with a smaller number of terminal nodes ($|T|$).

Cost complexity pruning

- **Big idea:** minimize RSS plus an additional penalty that rewards small **trees**

$$\overbrace{\sum_{m=1}^{|T|} \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2}^{\text{RSS}} + \overbrace{\alpha |T|}^{\text{Shrinkage penalty}}$$

- **Fun fact:** as we increase α , branches get pruned in a nice, predictable (nested) fashion (why is this useful?)

Tree variation of backward selection

Start by growing some big tree on the training data

1. Use cost complexity pruning to get a sequence of “best subtrees” (as a function of α)
2. Select a single “best” α using cross-validated prediction error or something similar
3. Return the associated tree

Discussion

- The minimization we just saw would help us find the best **regression** tree, but our example was about **classification**
- **Question:** what needs to change?

$$\sum_{m=1}^{|T|} \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha |T|$$

- **Answer:** just like in previous classification settings, we can't use RSS

Trouble in paradise...

- Usual approach (minimizing classification error) isn't sensitive enough to **build** good trees
- Alternative 1: *Gini index of each node*

$$G = \sum_{k=1}^K \hat{p}_{mk}(1 - \hat{p}_{mk})$$

- Alternative 2: *cross-entropy of each node*

$$D = \sum_{k=1}^K \hat{p}_{mk} \log \hat{p}_{mk}$$

- Both are measures of **purity***

*small values \rightarrow node contains mostly observations from the same class

Discussion

What advantages / disadvantages might decision trees have when compared to other methods?

Activity

Use the Hitters dataset (in the ISLP package)

Consider these variables:

- Years, Hits, RBI

We will use these three variables to create a decision tree that predicts Salary.

Start by dropping any players with NA salary.

Then, decide which variable to use for your first split, and what value of that variable to split around.

Then repeat until you have 4-6 terminal nodes.

Draw your decision tree.