Recap

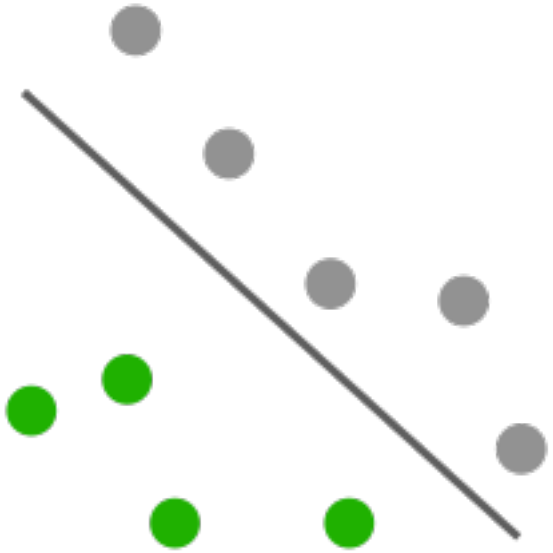
Families of ML algorithms

- Linear
- Tree-based
- kNN
- Neural Networks

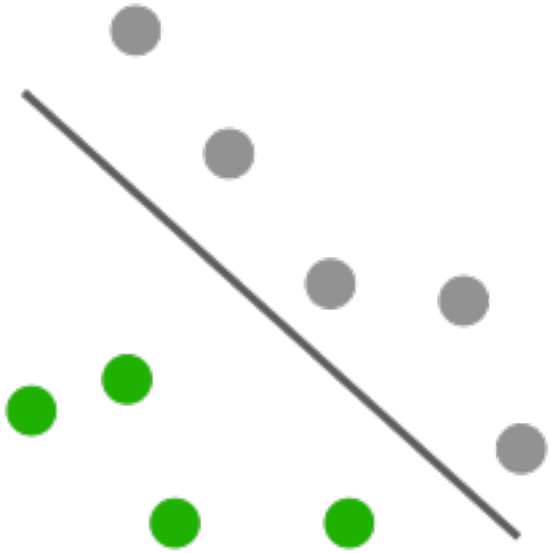
Linear model

Linear model

Linear model



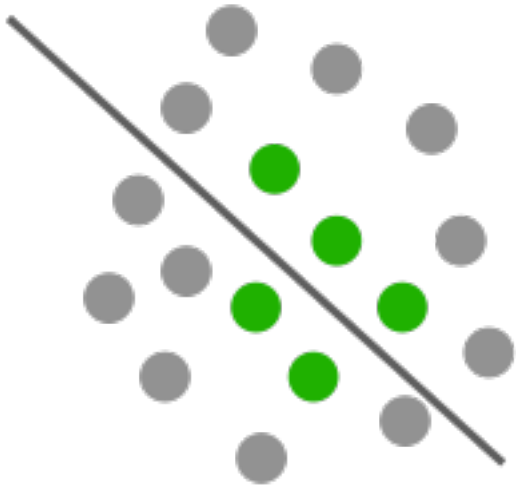
Linear model



Examples:

- Logistic Regression
- Support Vector Machines

Linear model

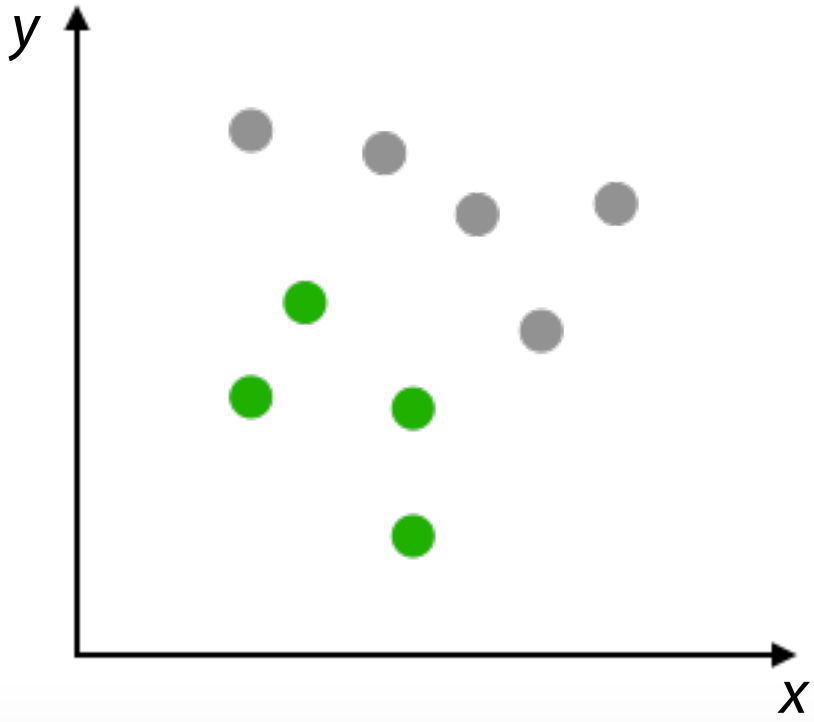


Linear model

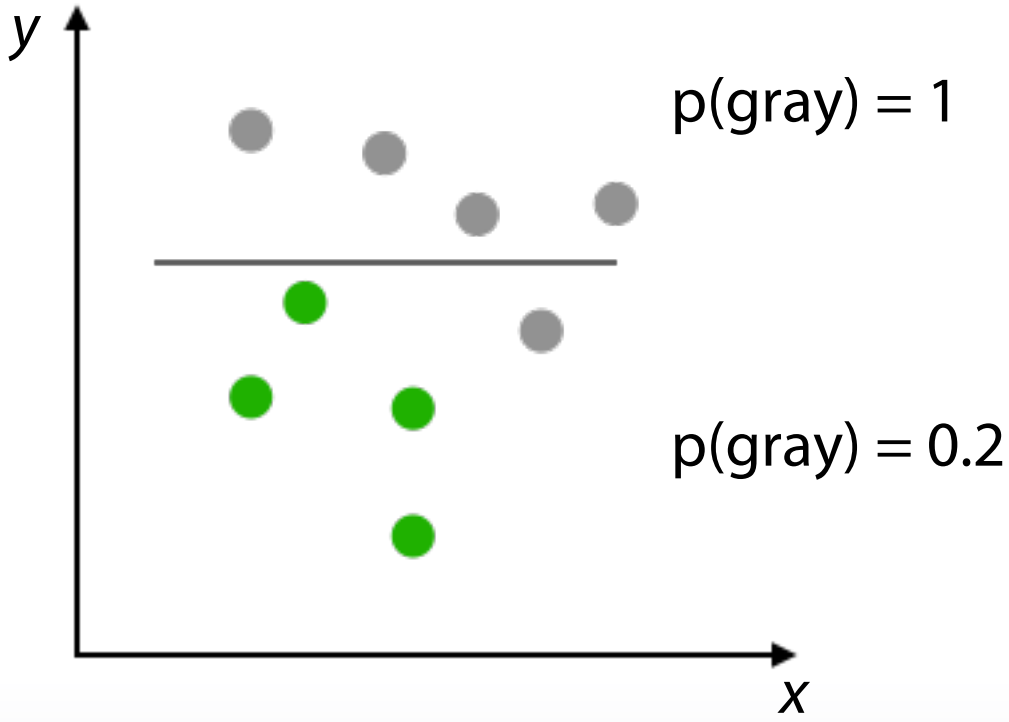
VOWPAL WABBIT

Tree-based: Decision Tree, Random Forest, GBDT

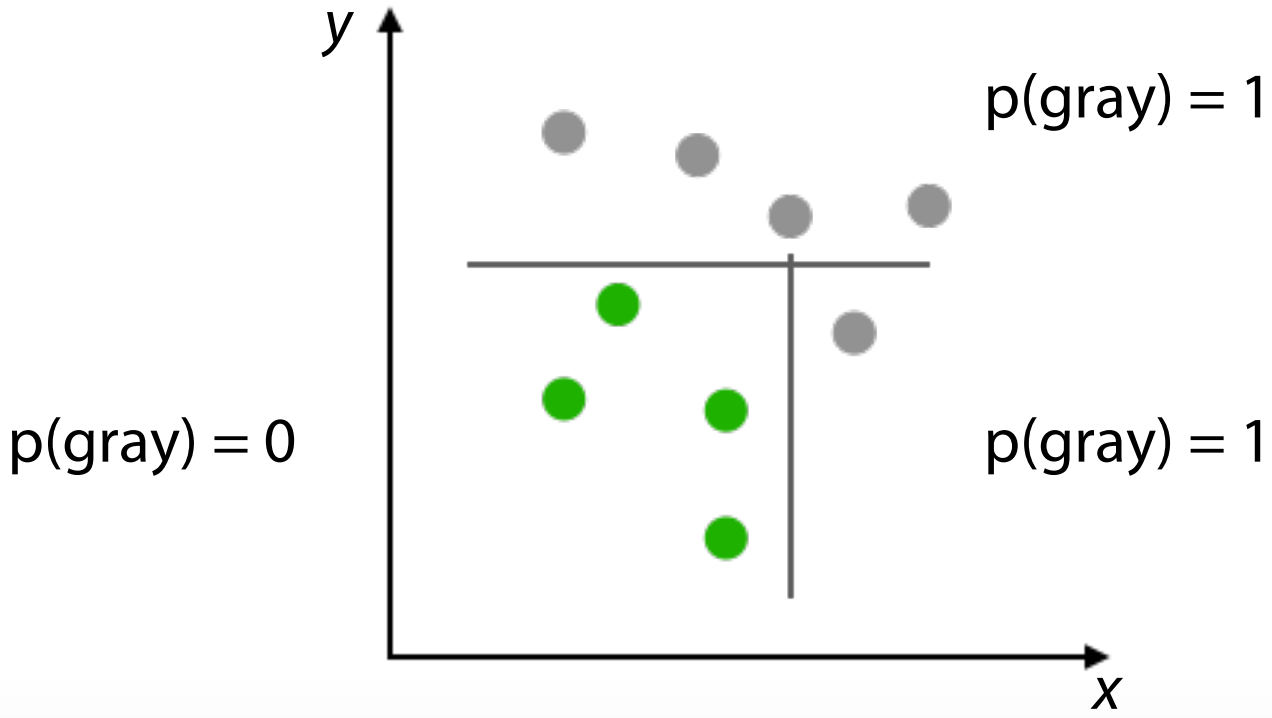
Tree-based: Decision Tree, Random Forest, GBDT



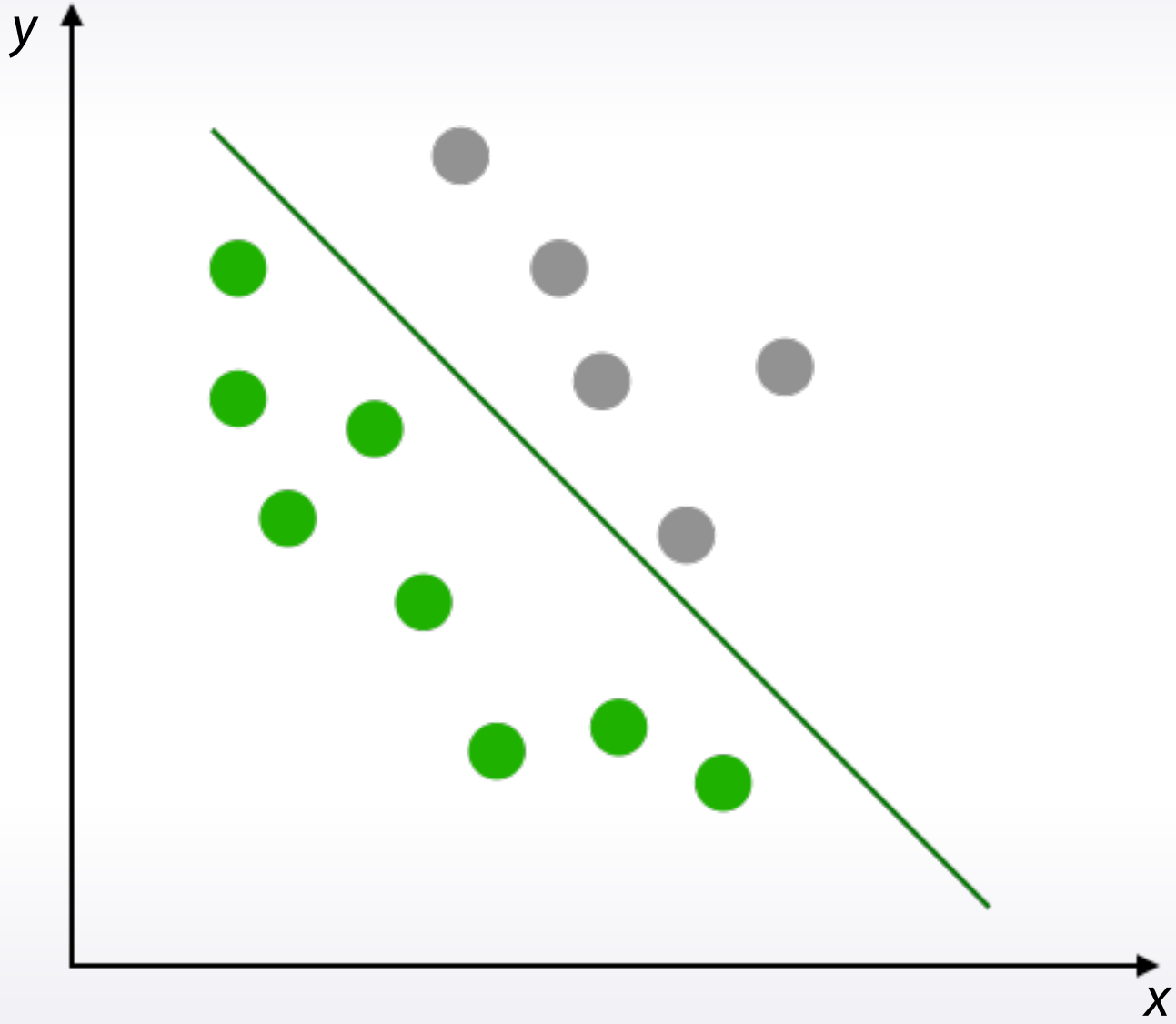
Tree-based: Decision Tree, Random Forest, GBDT



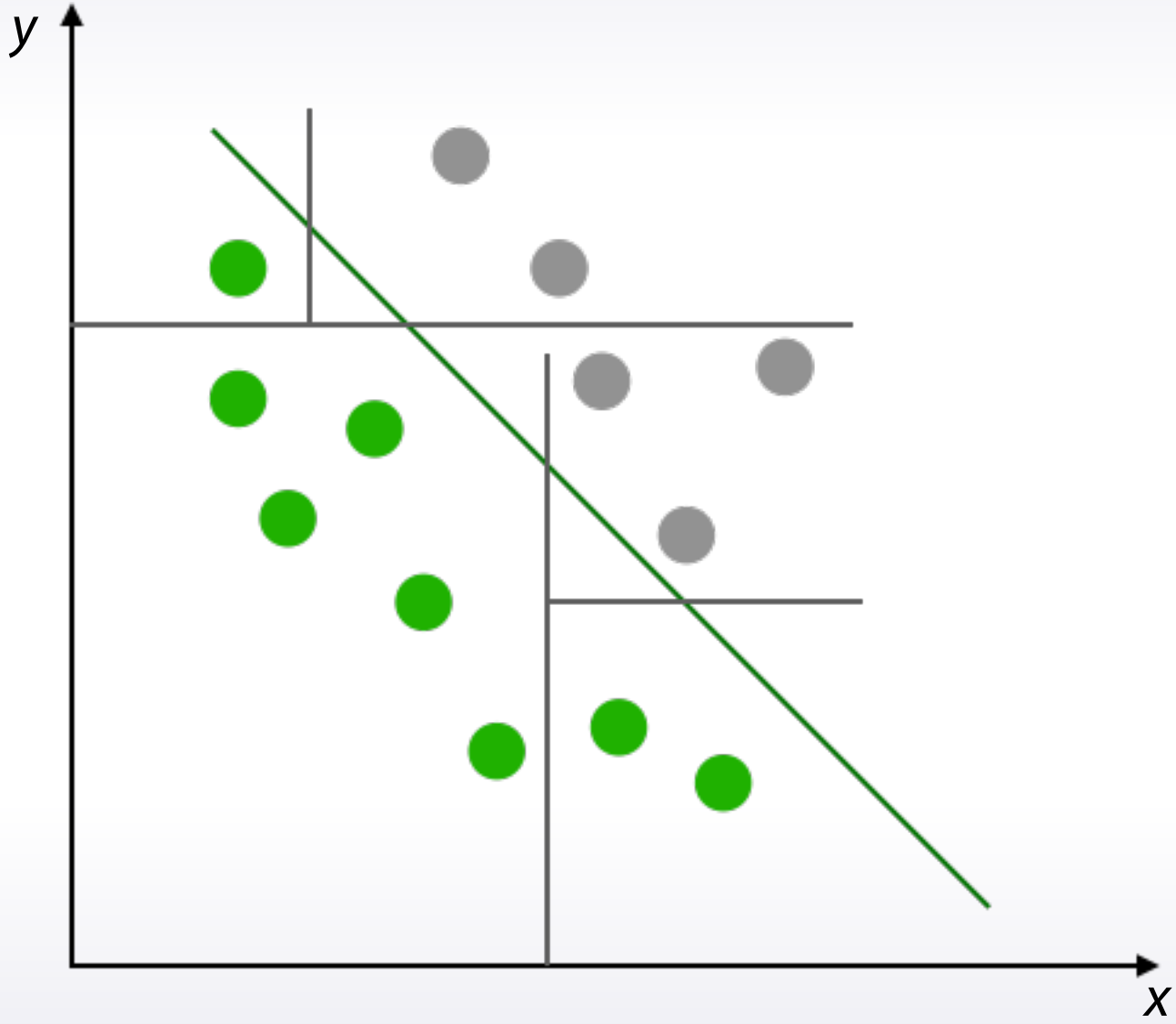
Tree-based: Decision Tree, Random Forest, GBDT



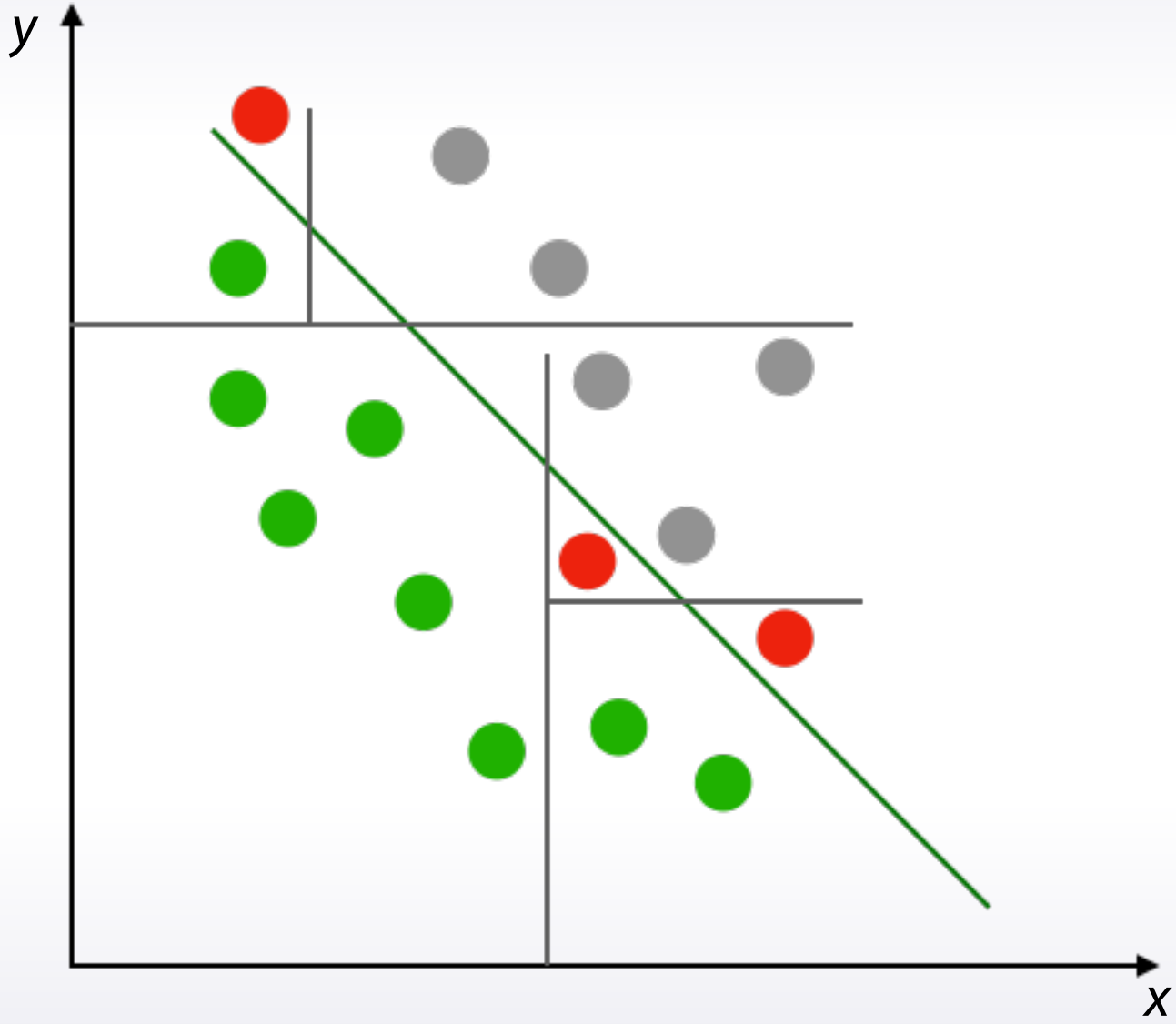
Tree-based: Decision Tree, Random Forest, GBDT



Tree-based: Decision Tree, Random Forest, GBDT



Tree-based: Decision Tree, Random Forest, GBDT

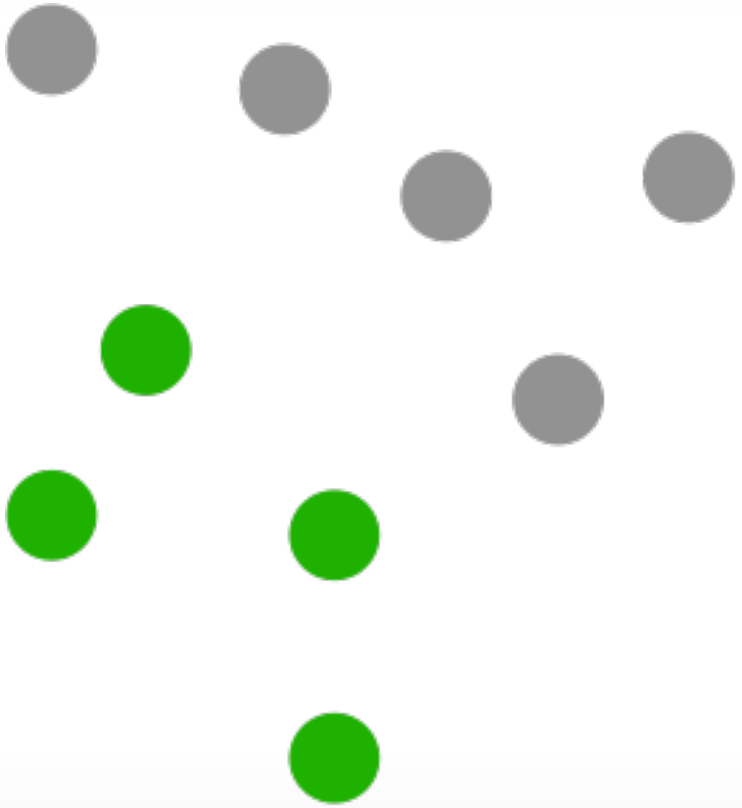


Tree-based methods

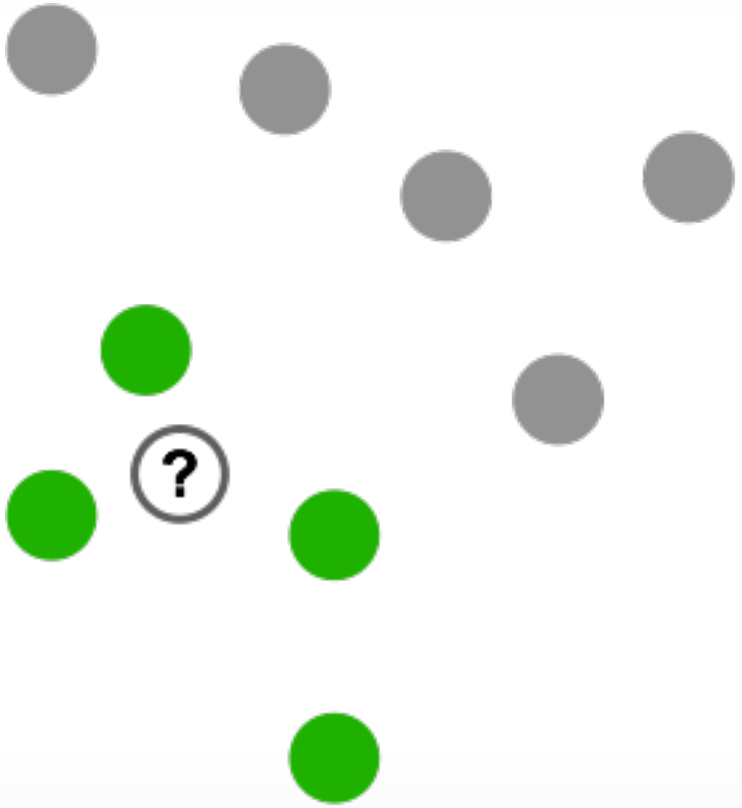
Microsoft / LightGBM

kNN-based methods

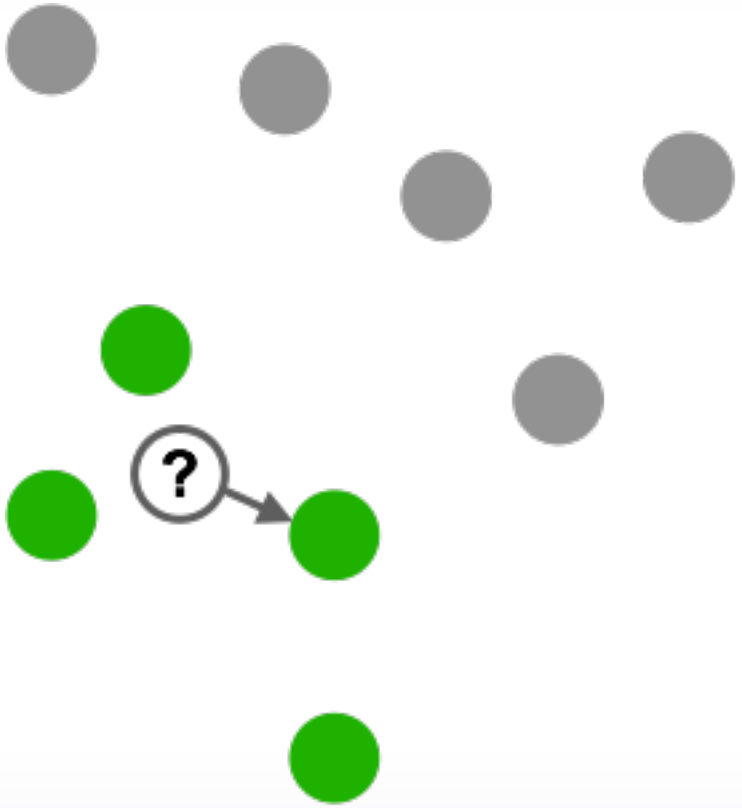
kNN-based methods



kNN-based methods

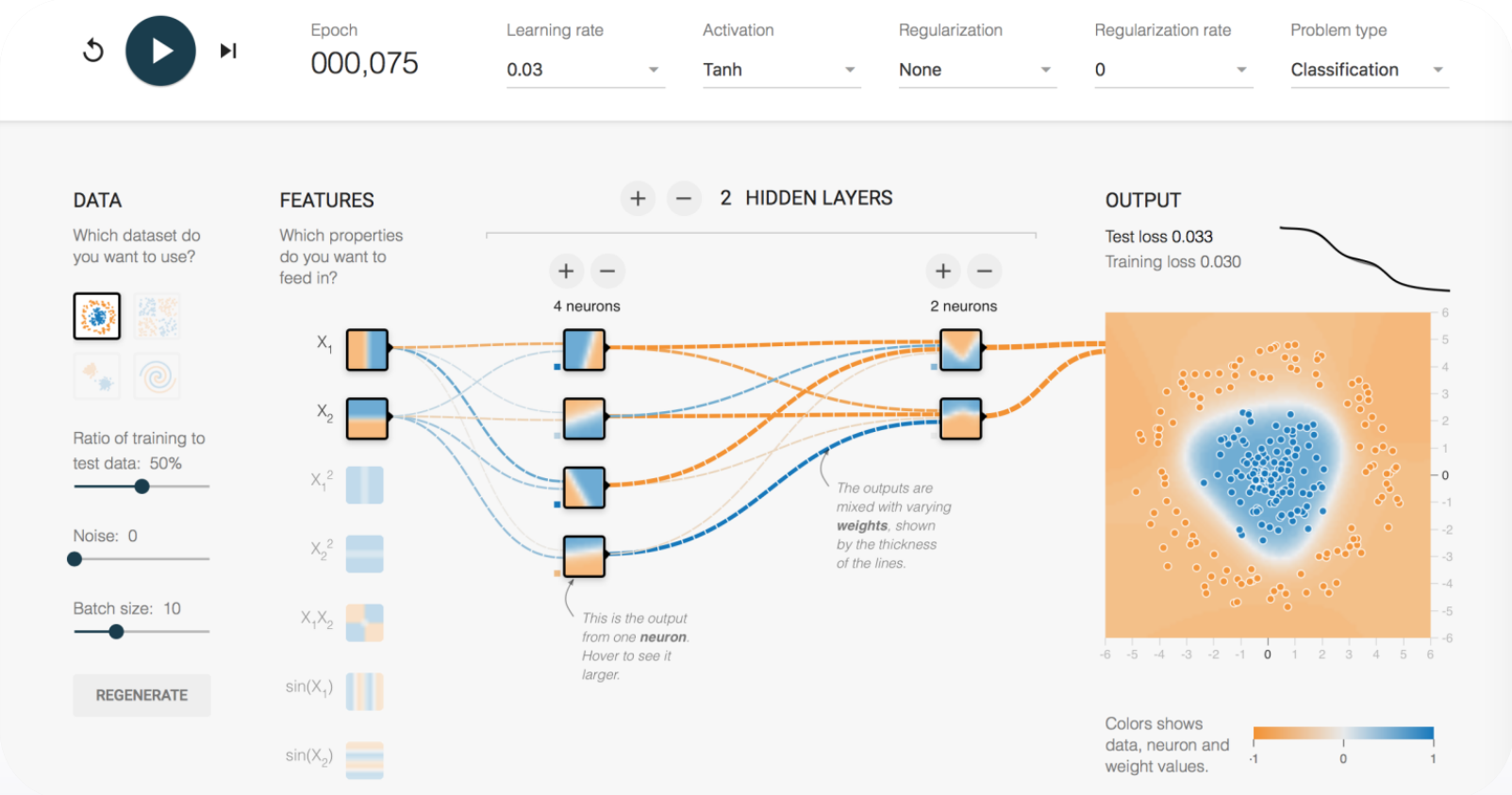


kNN-based methods



kNN-based methods

Neural Networks



Tensorflow Playground, http://playground.tensorflow.org

Neural Networks

No Free Lunch Theorem

No Free Lunch Theorem

"Here is no method which **outperforms all others for all tasks**"

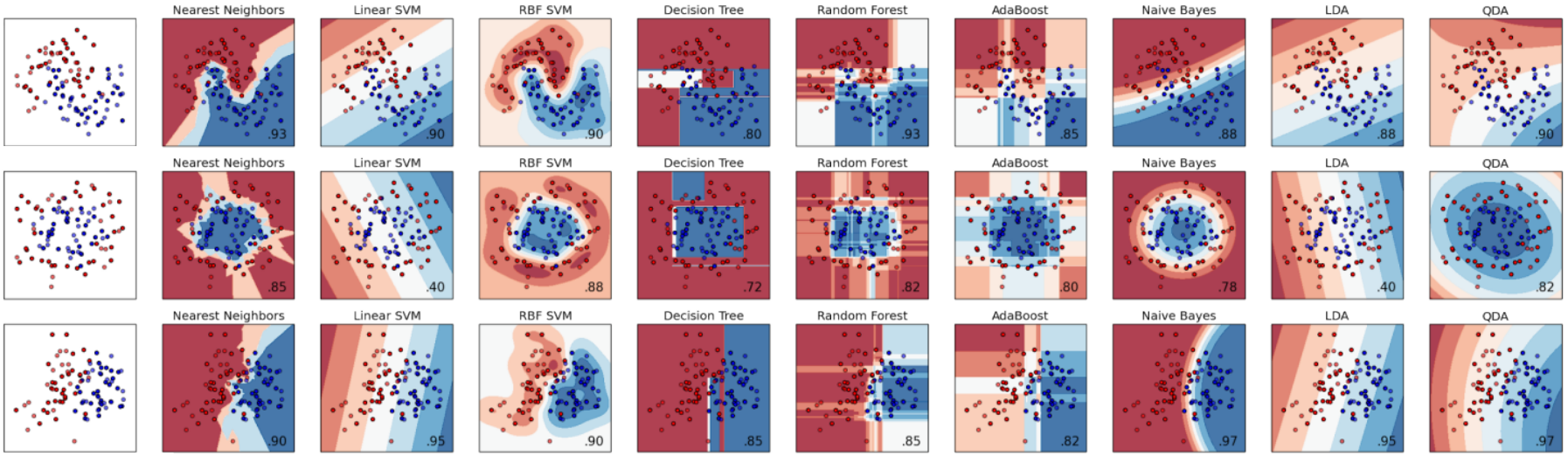
No Free Lunch Theorem

"Here is no method which **outperforms all others for all tasks**"

or

"For every method **we can construct a task** for which **this particular method will not be the best**"

Decision surfaces



Classifier comparison, http://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html

Conclusion

- There is no “silver bullet” algorithm

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes
- k-NN methods heavy rely on how to measure points “closeness”

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes
- k-NN methods heavy rely on how to measure points “closeness”
- Feed-forward NNs produce smooth non-linear decision boundary

Conclusion

- There is no “silver bullet” algorithm
- Linear models split space into 2 subspaces
- Tree-based methods splits space into boxes
- k-NN methods heavy rely on how to measure points “closeness”
- Feed-forward NNs produce smooth non-linear decision boundary

The most powerful methods are
Gradient Boosted Decision Trees and **Neural Networks**.
But you shouldn’t underestimate the others