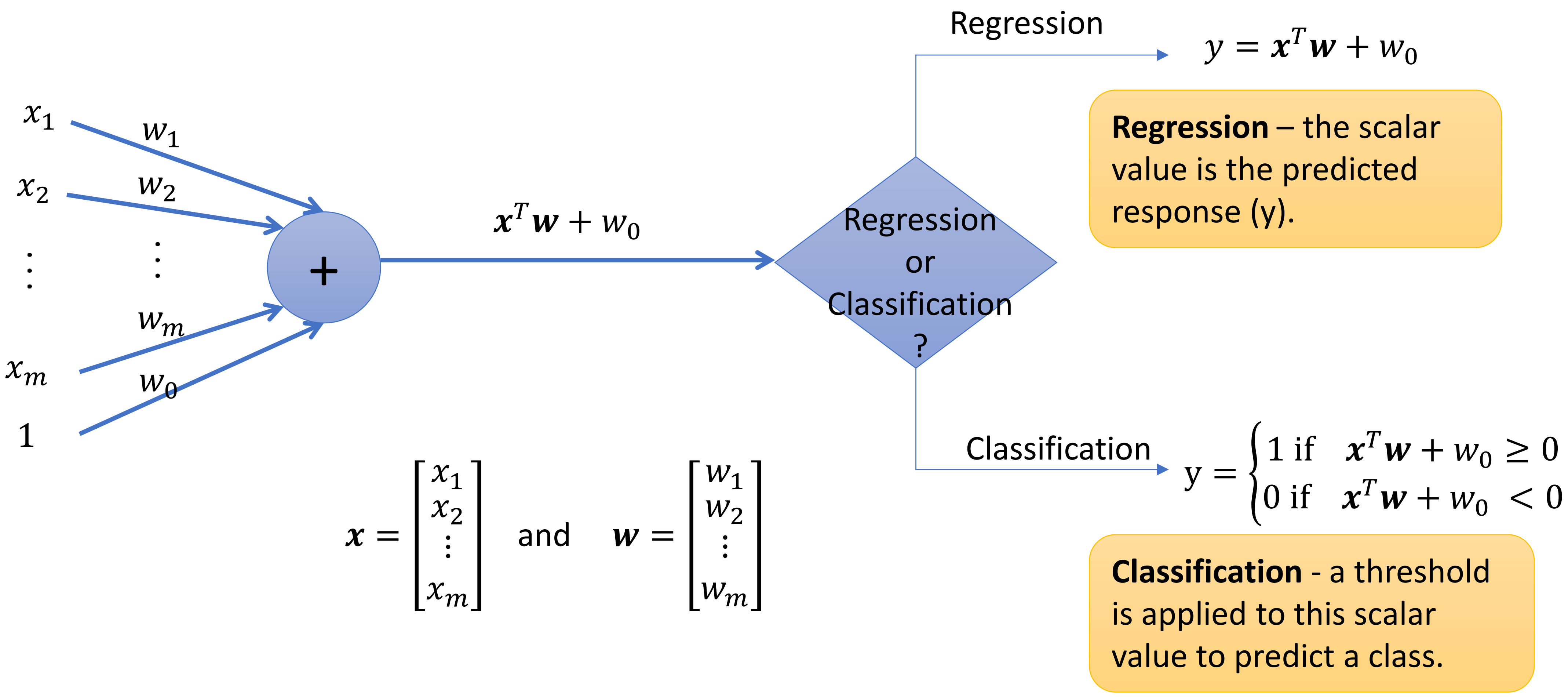
Non-Linear Models & Decision Boundaries

CSC4601 Theory of Machine Learning

Plan

- Advantages and disadvantages of Linear Models
- Extending linear models
- Example of non-linear models: focus on showing model’s response (regression) & decision boundaries (classification)
- Simple vs Complex model: in terms of model’s response (regression) & decision boundaries (classification)
- Effect of feature scaling on the shape of decision boundaries

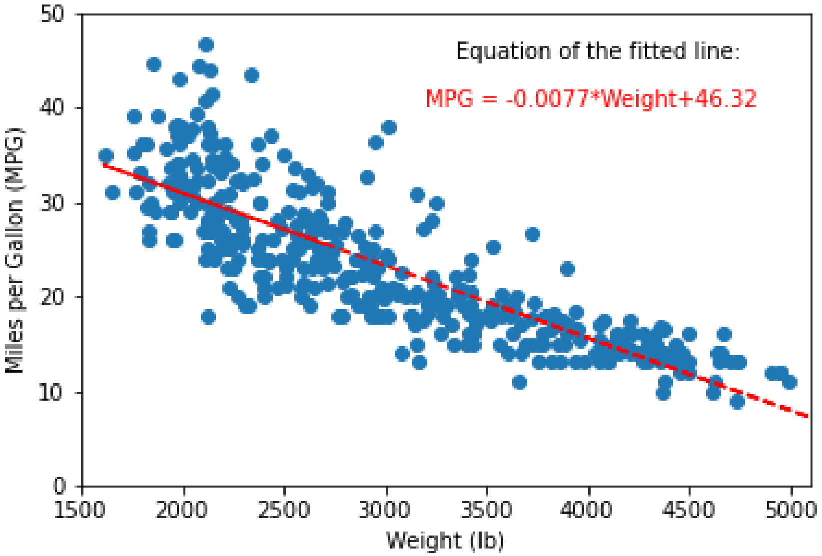
Regression and Classification Linear model



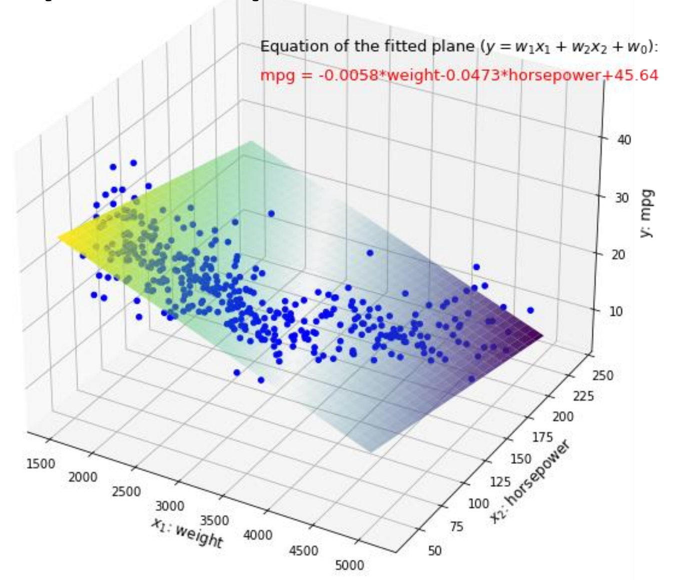
Linear Regression Models

- They assume a linear relationship between the features of a sample and the label.

Straight-line relationship (1 feature)



With two features, the linear model's response is a plane.

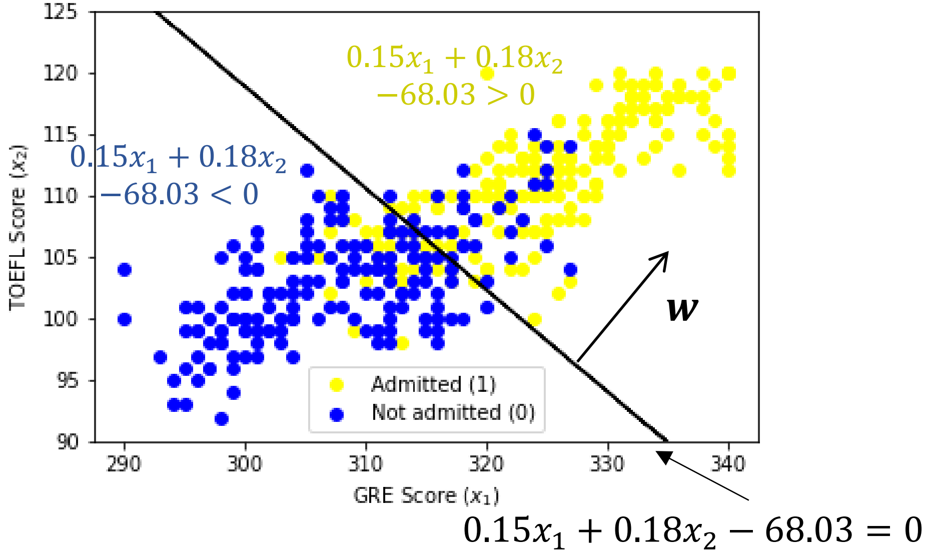


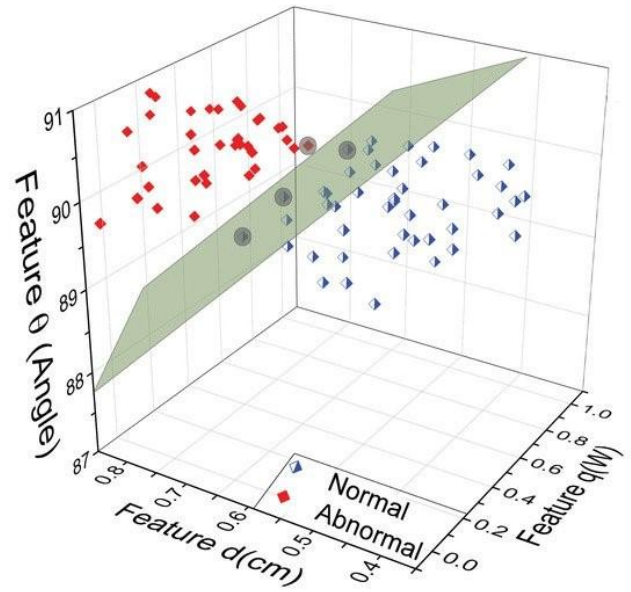
With more than two features, the shape of model's response: hyperplane

Linear Classification Models

- Linear classification models (like linear SVM and logistic regression) divides the feature space into two half-spaces: each half-space is a decision region.
- The decision regions are separated by a hyperplane.

A line divides/partitions the 2-dimensional feature space





- **A plane divides/partitions the 3-dimensional feature space**
- **A hyperplane divides the m-dimensional feature space.**

Linear Models

Advantages

Disadvantages

Linear Models

Advantages

- Simple model
- When used for prediction, they are computationally fast. Since they summarize data with a finite set of weights, storing trained linear models only requires storing the weights.
- Linear model are interpretable: the weights helps us understand the contribution of each feature to the overall prediction.

Disadvantages

Linear models has limitations. They can be too simple:

- What if classes are not linearly separable ? (classification)
- What if features are not linearly related to the response?

Limitations of Linear Models

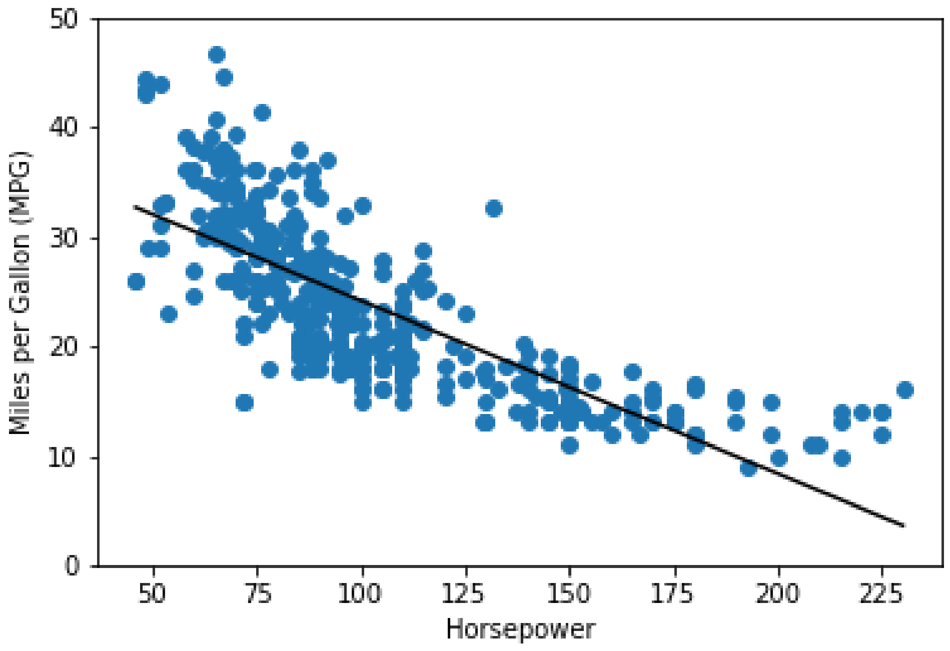
- In regression, the true relationship between the response and features might be non-linear.
- In classification, classes might not be linearly separable.
- Linear models model a straight-line relationship, which might not be flexible enough to model more complex relationship.
- This is why non-linear models are said to be **more complex or flexible** than linear models.

Extending Linear Models

- Linear models can be extended to accommodate non-linear relationships.

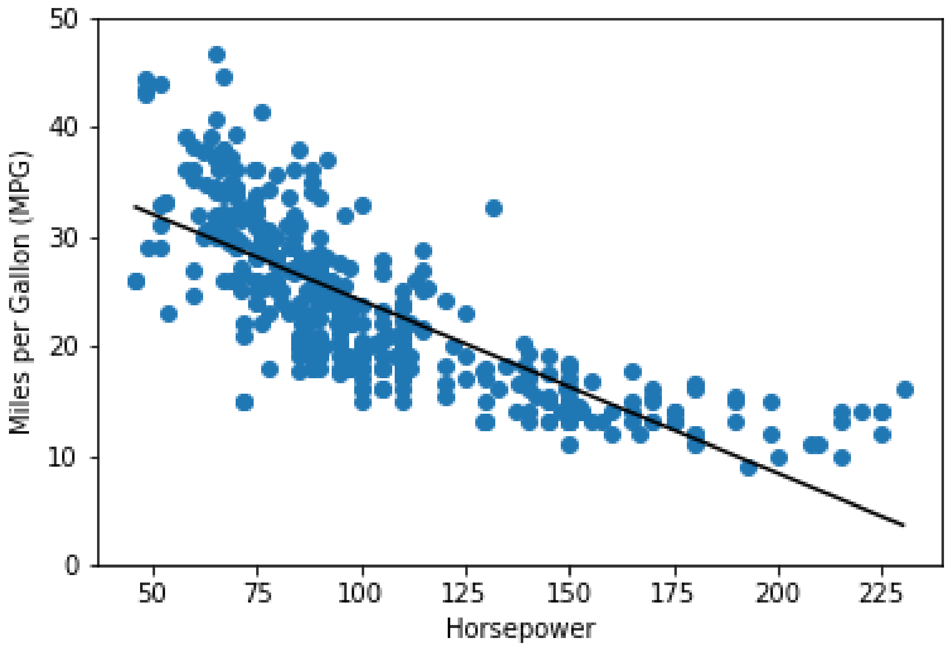
- The very simple way: extend the feature space by adding polynomial features to the existing features in the data.

Extending Linear Models - Regression



- We can see that the relationship between mpg and horsepower is in fact a curved relationship.
- Fitting the data with a straight line might not be the best approach. Can we do better?

Extending Linear Models - Regression



- We can see that the relationship between mpg and horsepower is in fact a curved relationship.
- Fitting the data with a straight line might not be the best approach. Can we do better?
- What if add an additional feature: horsepower^2

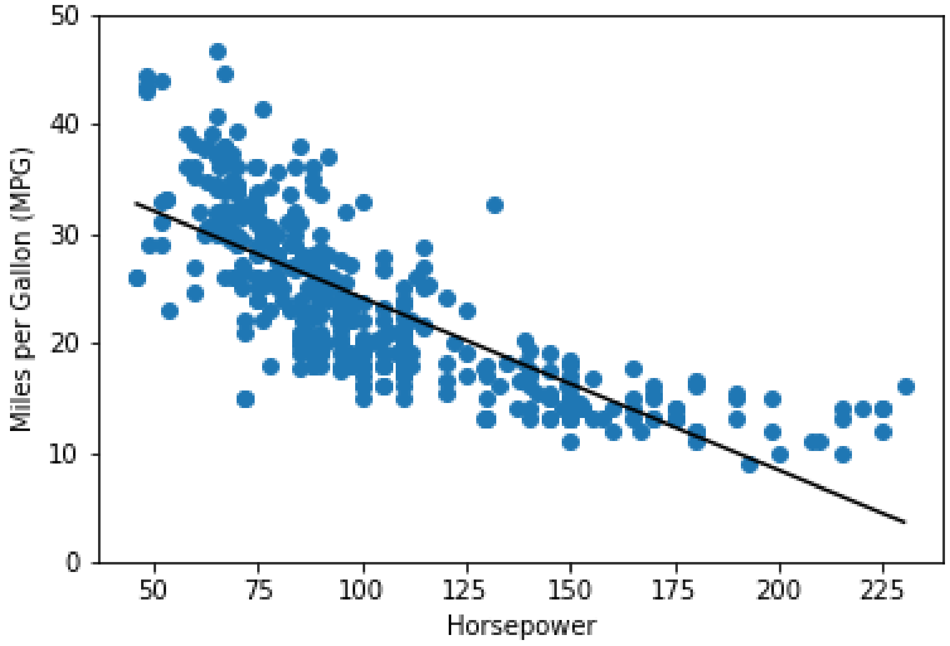
So instead of fitting the line:

$$\text{mpg} = w_0 + w_1 \times \text{horsepower}$$

we fit the curve:

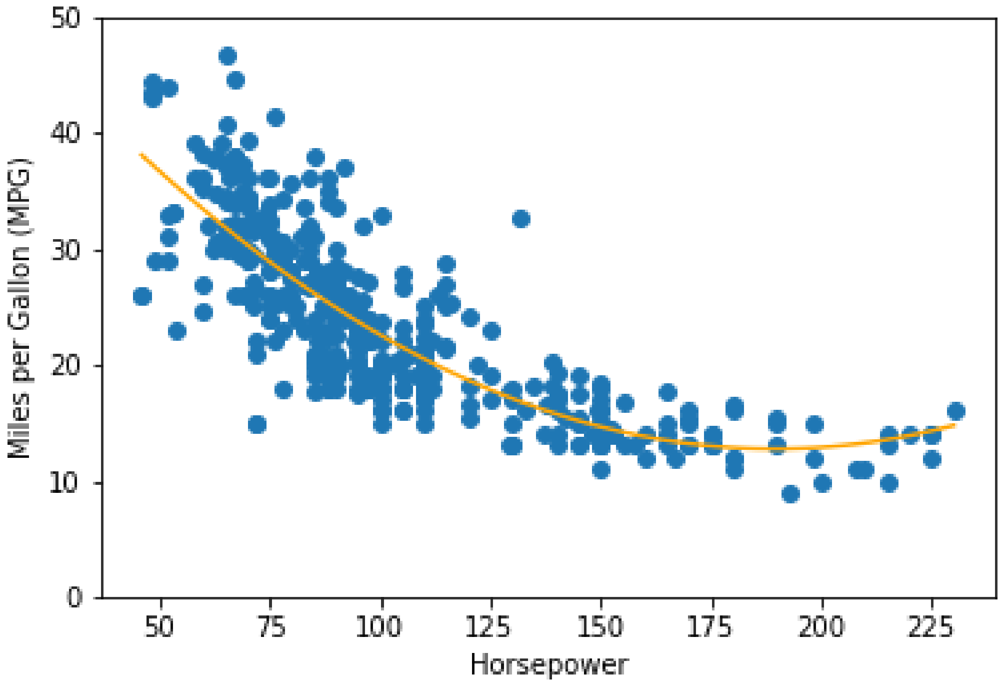
$$\text{mpg} = w_0 + w_1 \times \text{horsepower} + w_2 \times \text{horsepower}^2$$

Extending Linear Models - Regression



Modeling relationship between mpg and horsepower as:

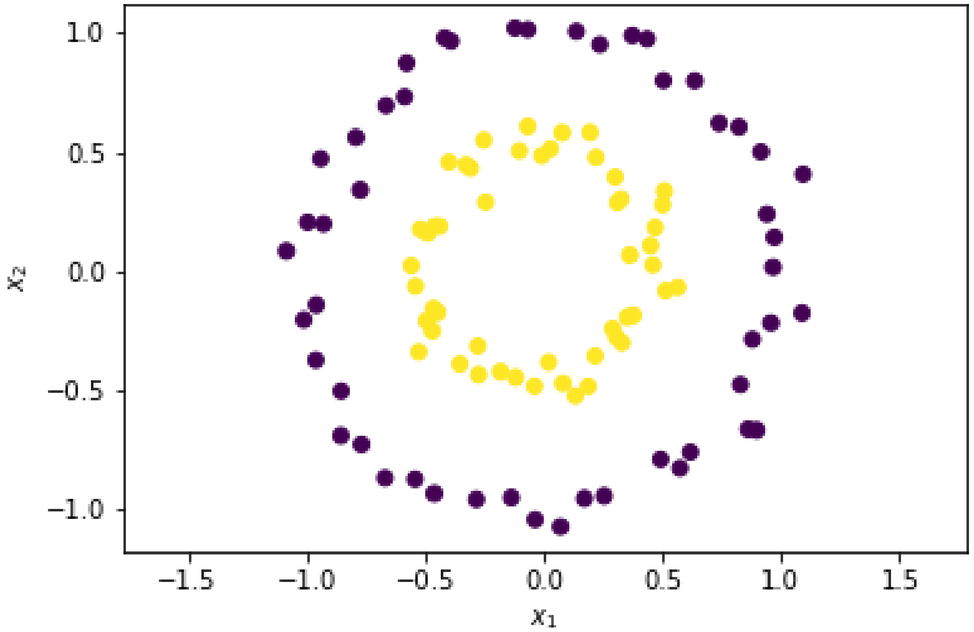
$$\text{mpg} = w_0 + w_1 \times \text{horsepower}$$



Modeling relationship between mpg and horsepower as:

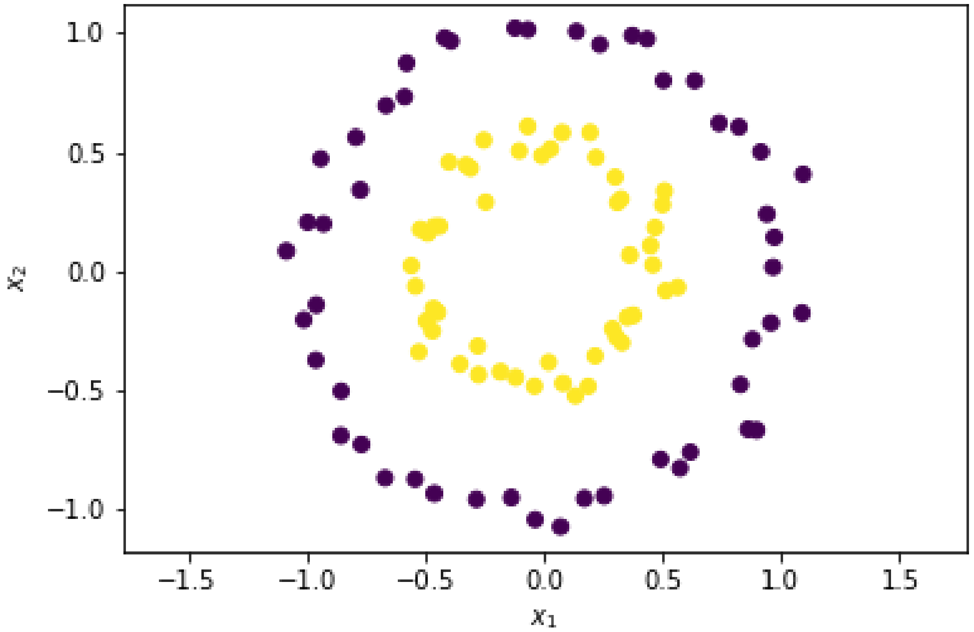
$$\text{mpg} = w_0 + w_1 \times \text{horsepower} + w_2 \times \text{horsepower}^2$$

Extending Linear Models - Classification



- We can see that two classes can't be separated with a line, however a circular decision boundary can perfectly separate the two classes.

Extending Linear Models - Classification



- We can see that two classes can't be separated with a line, however a circular decision boundary can perfectly separate the two classes.

So instead of finding the separating line,

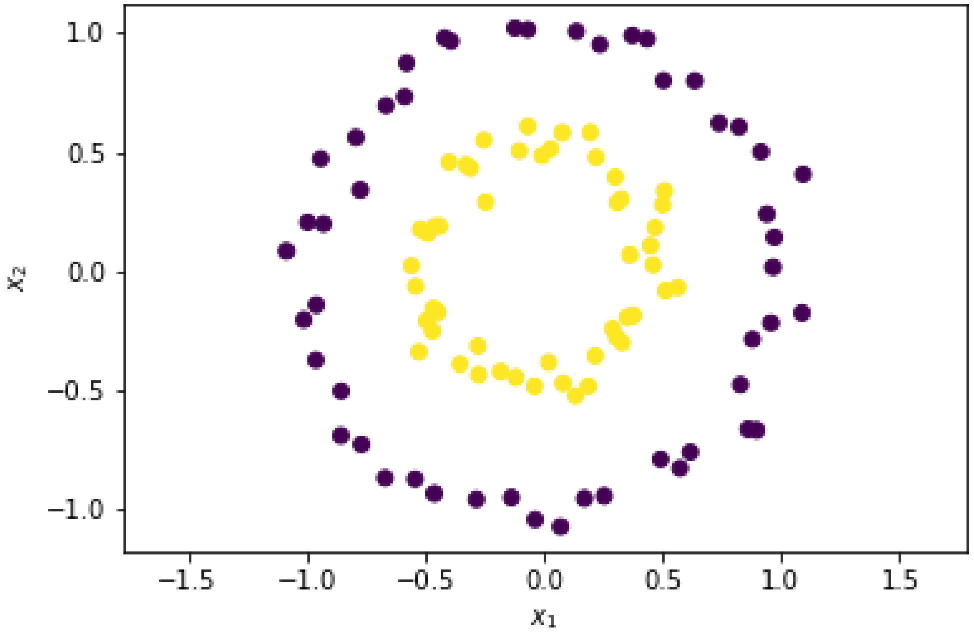
$$w_1 x_1 + w_2 x_2 + w_0 = 0$$

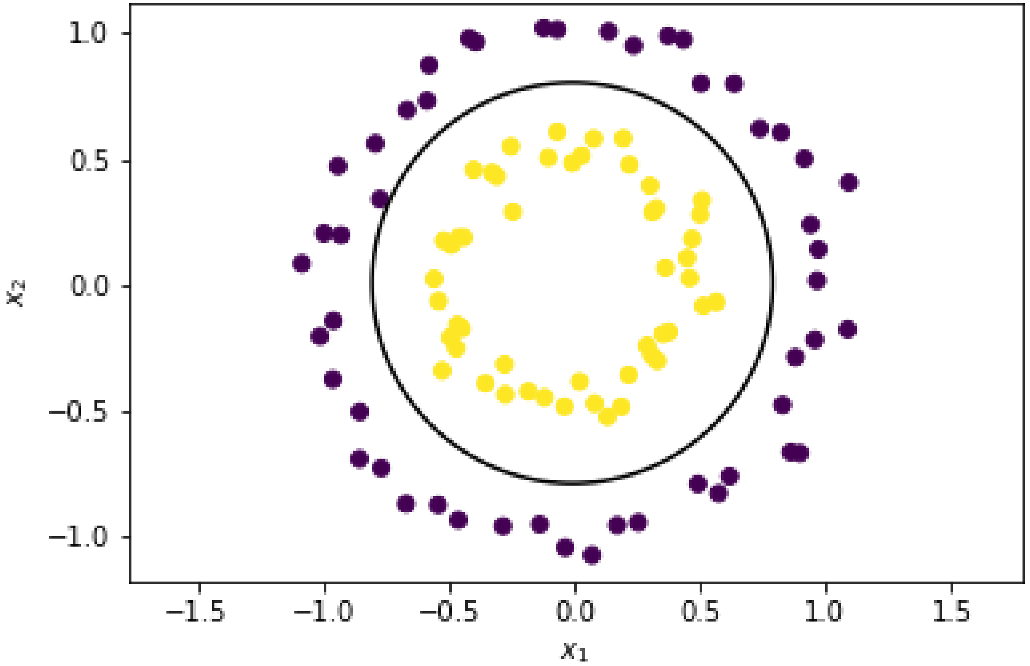
let’s try to find the circle

$$w_1 x_1 + w_2 x_2 + w_3(x_1^2 + x_2^2) + w_0 = 0$$

that can separate the two classes.

Extending Linear Models - Classification





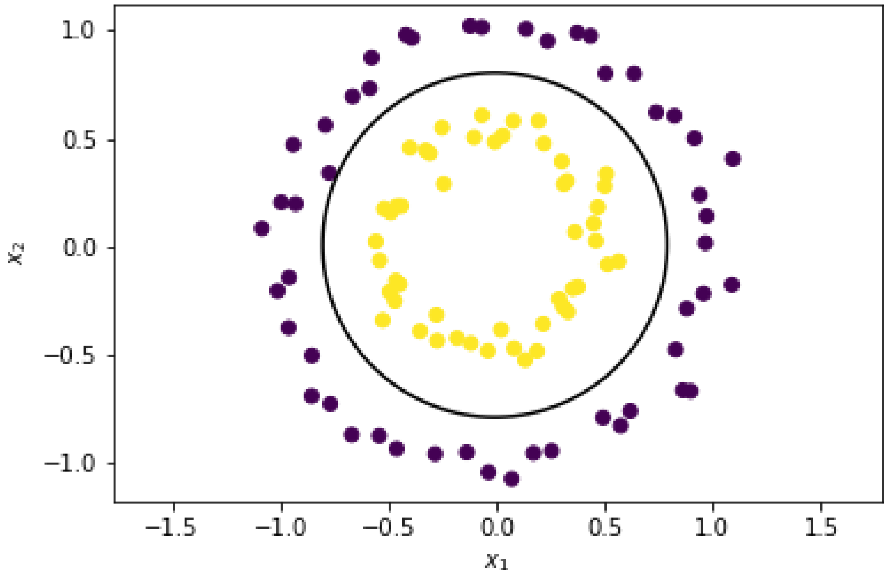
The linear classification model can be extended to separate the two classes with a circular decision boundary.

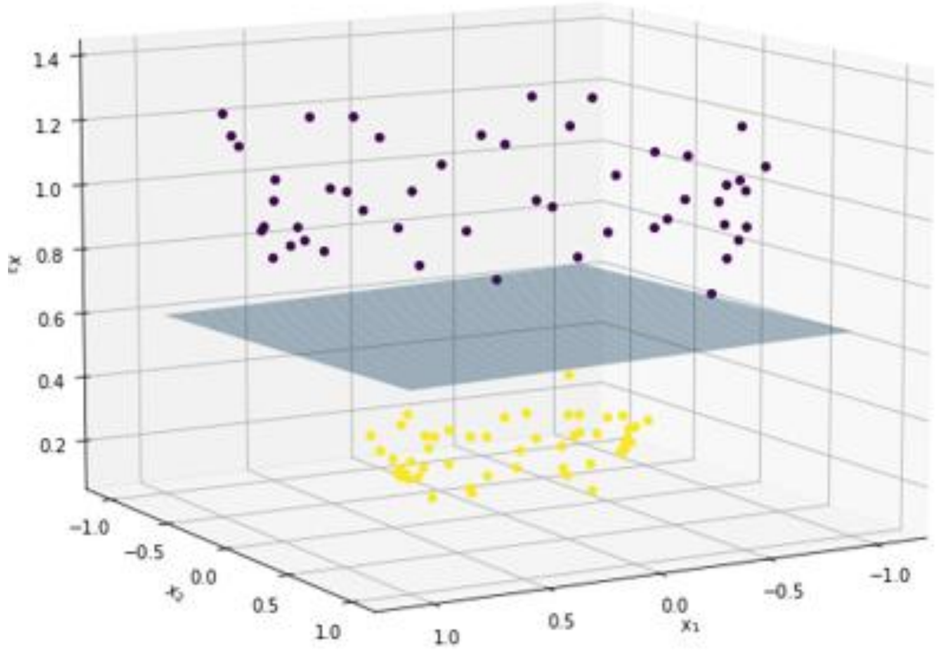
Extending Linear Models - Classification

- We can treat $x_1^2 + x_2^2$ as a third feature, so the model can be seen as a linear classification model with 3 features.

$$w_1 x_1 + w_2 x_2 + w_3(x_1^2 + x_2^2) + w_0 = 0$$

- The idea here is that by adding this additional feature, we're transforming the data from low to high dimensional space (from 2D to 3D) where it is easy for a linear model to separate the data.





Extending Linear Models

- Adding polynomial features:

  $(x_1, x_2, x_3, \cdots) \longrightarrow$

- There are other methods that make linear models accommodate for non-linear relationship:
  - Kernel trick with SVM
  - Spline regression

Non-Linear Models

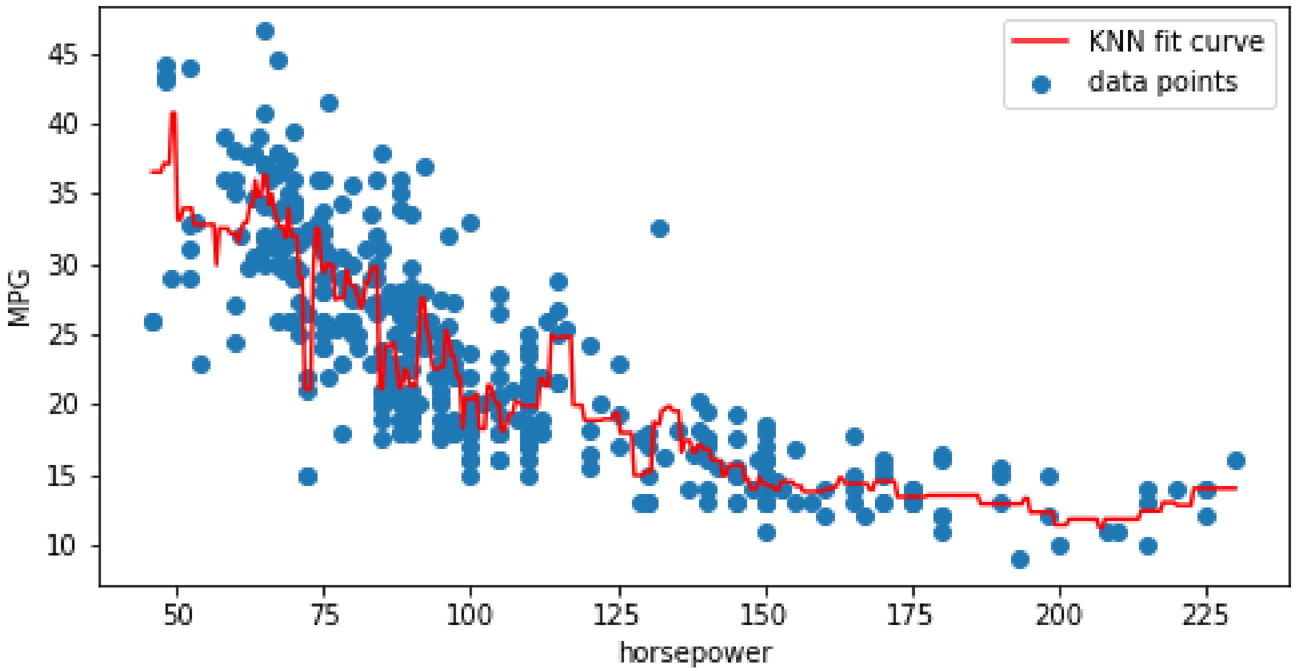
Example of non-linear models:

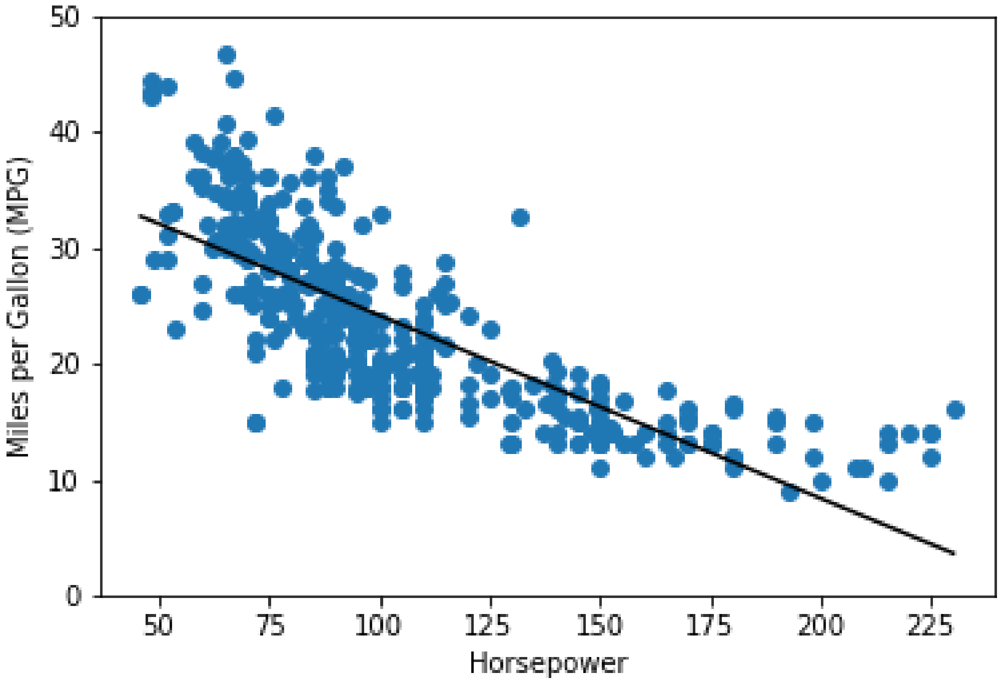
- K-nearest neighbor
- Decision trees
- Random forest
- Boosted trees

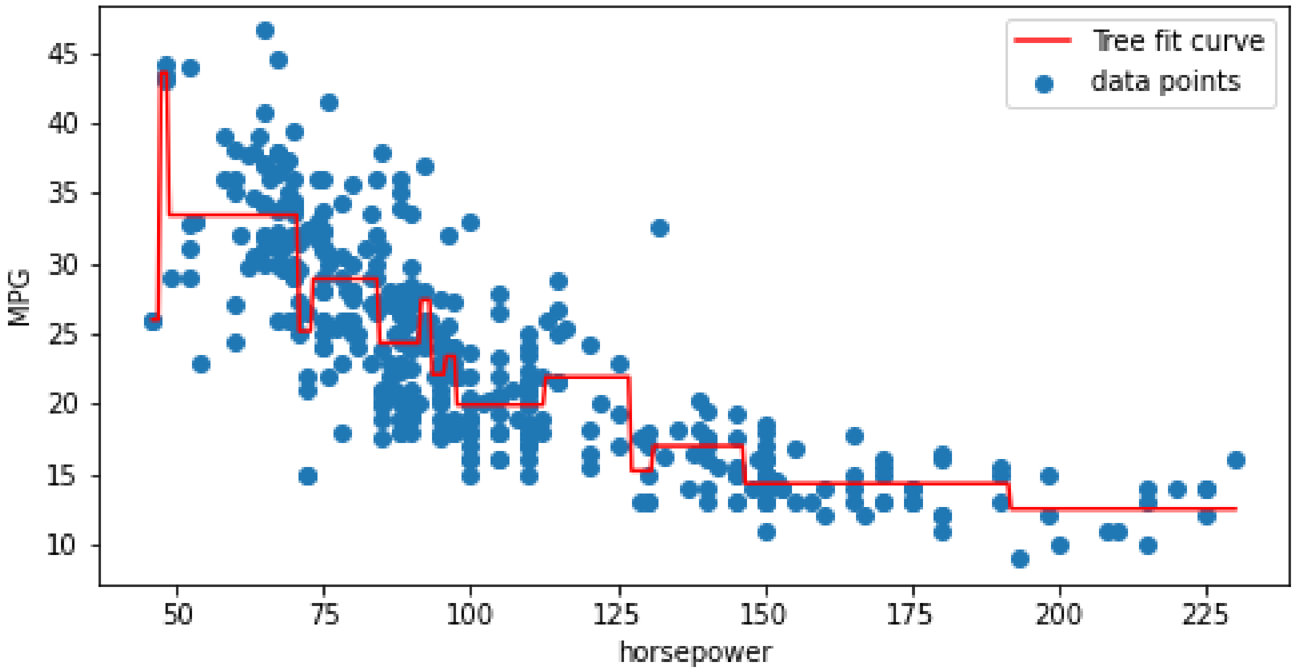
In the next slides, we will show examples of model’s response or decision boundaries of KNN and decision tree models.

Example of non-linear regression models

- KNN and Trees are more complex models than linear model: shape of model's response is more flexible!



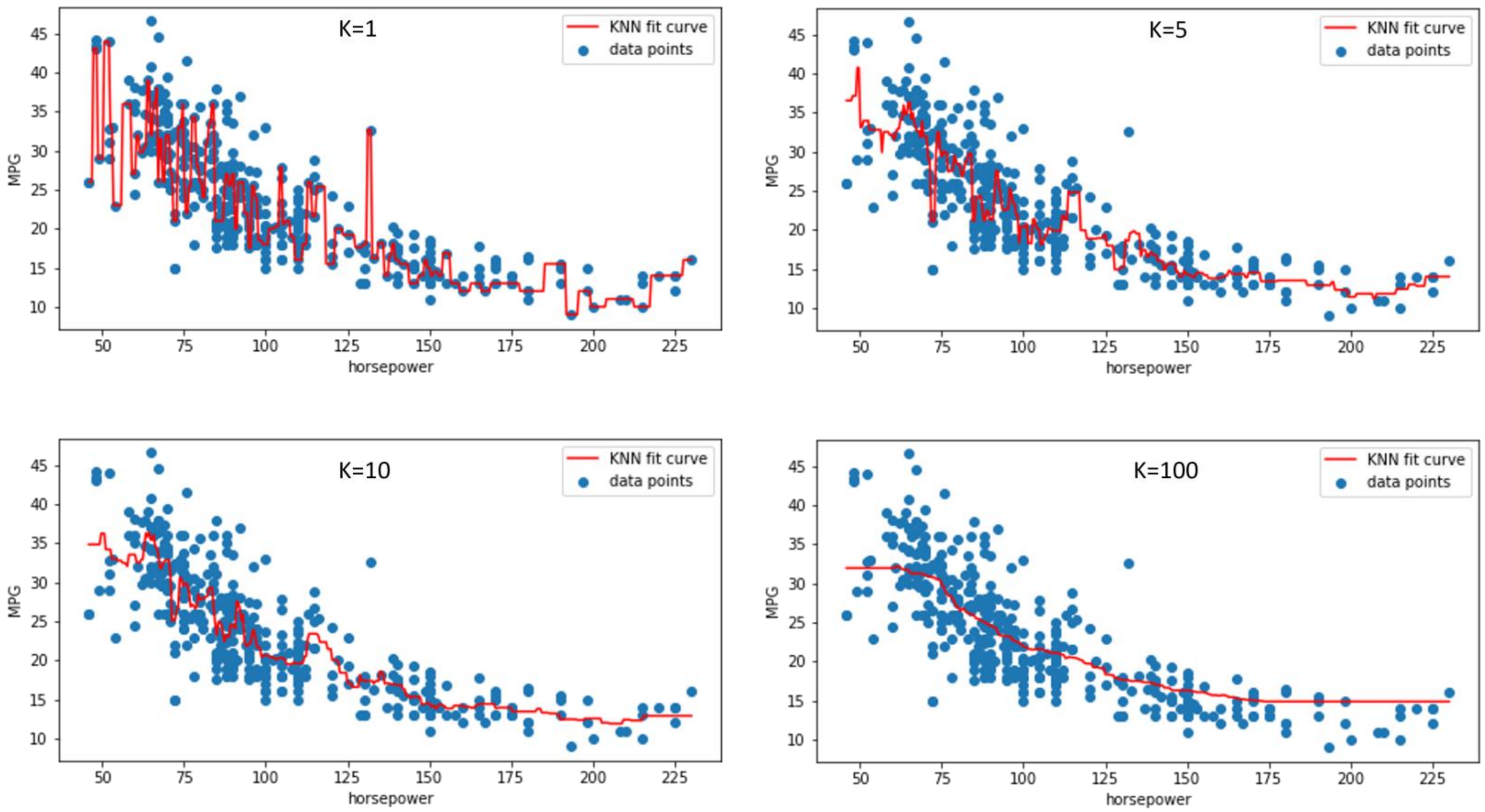




Example of non-linear regression models

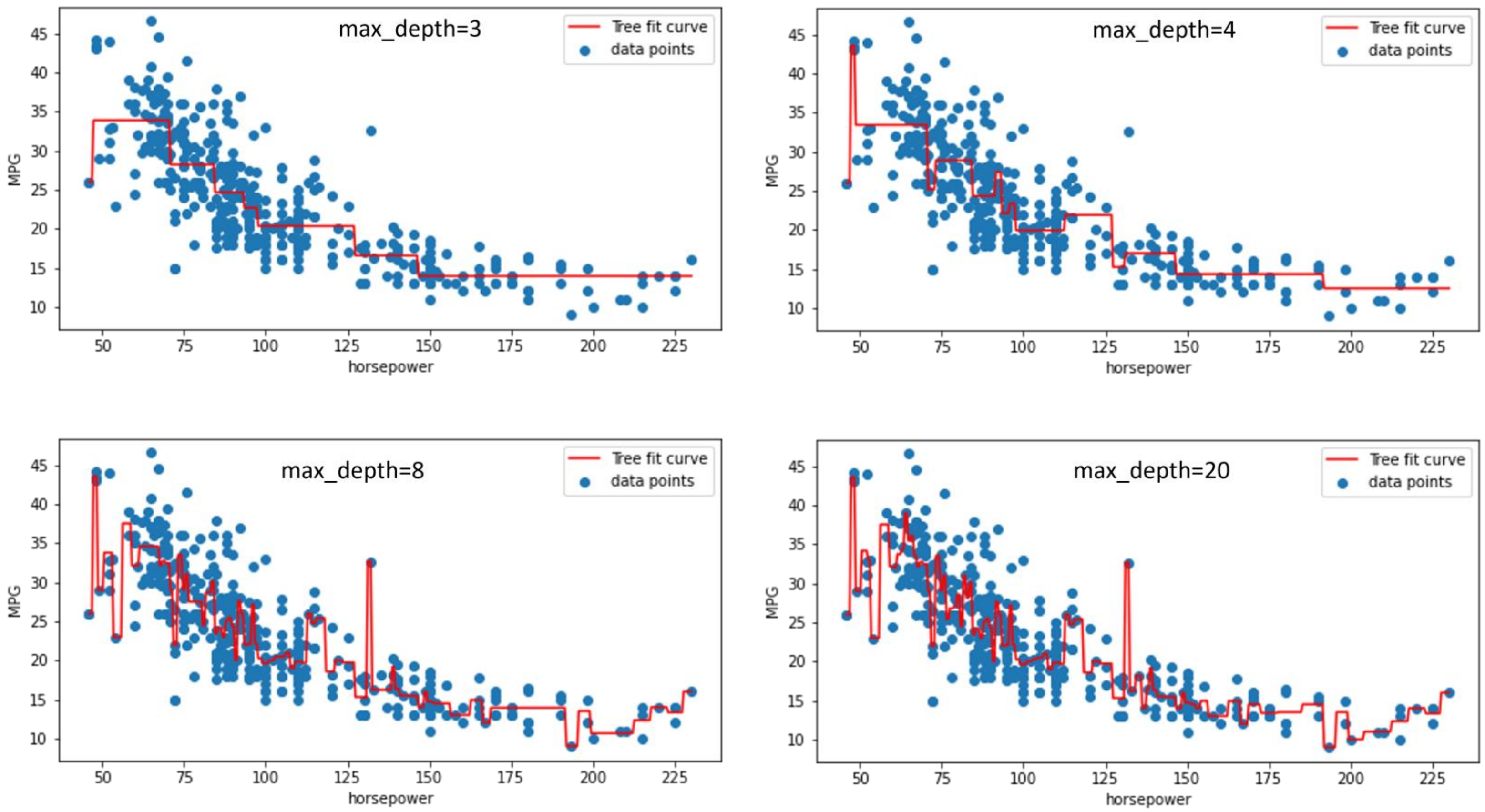
- KNN and Trees have extra settings (hyperparameter) that can be changed to change the complexity of the model itself.

- How does the complexity of KNN change with changing K?



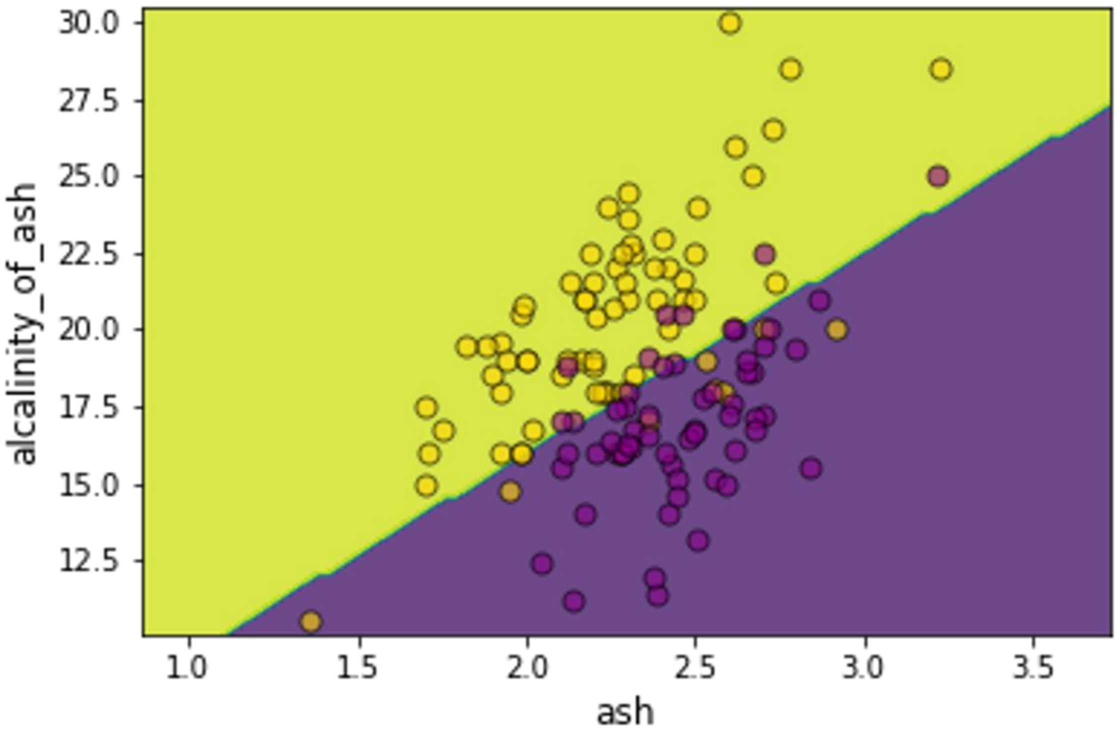
Example of non-linear regression models

- KNN and Trees have extra settings (hyperparameter) that can be changed to change the complexity of the model itself.
- How does the complexity of trees change with changing max_depth?

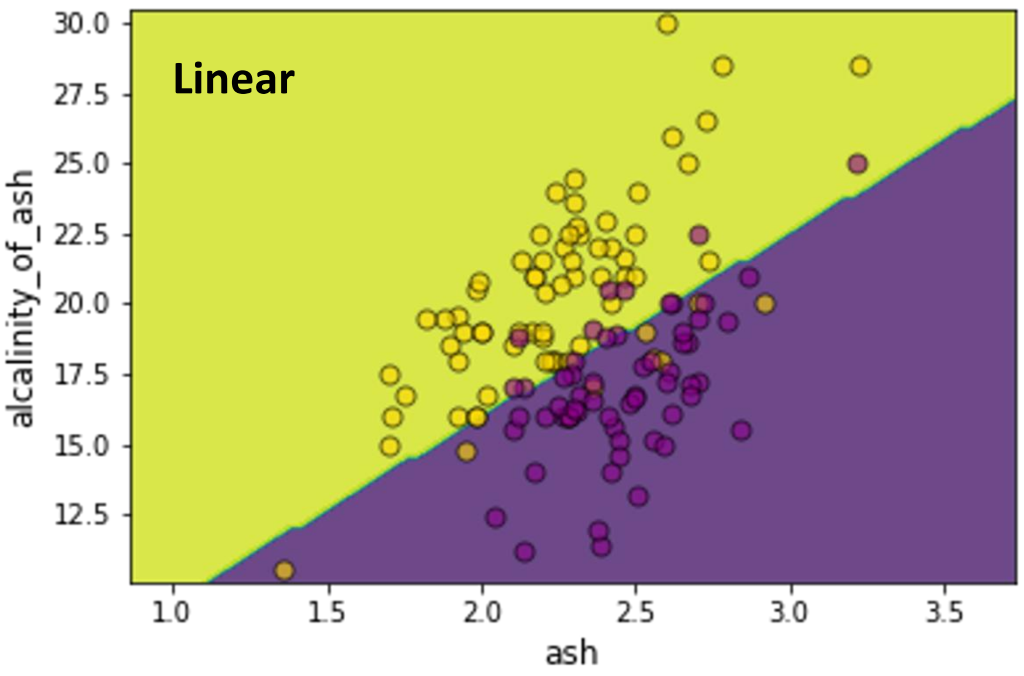


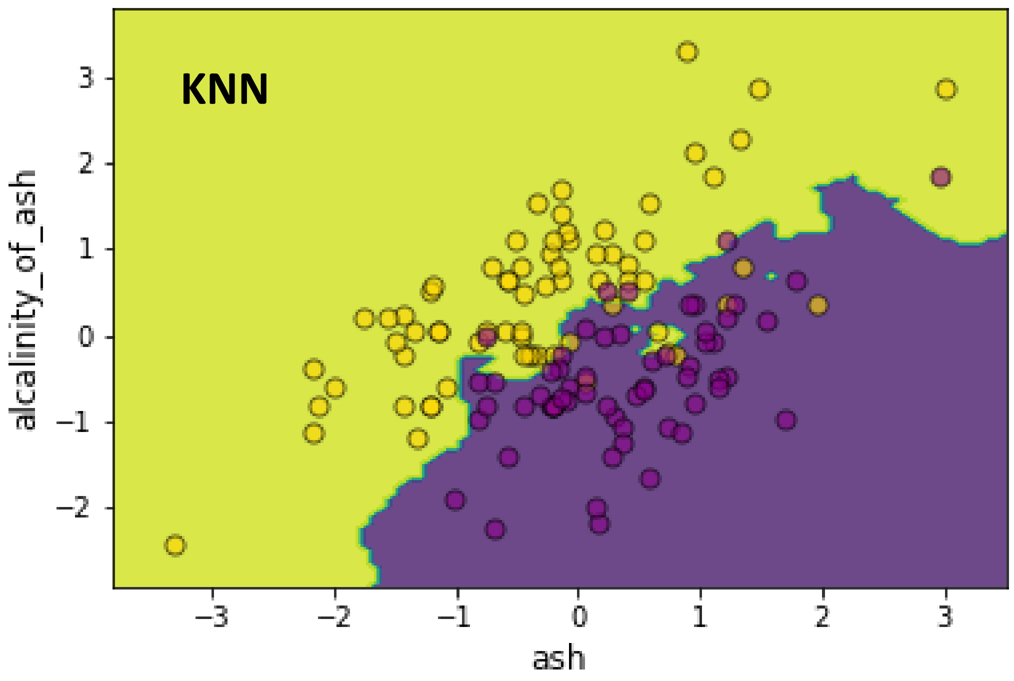
Example of non-linear classification models

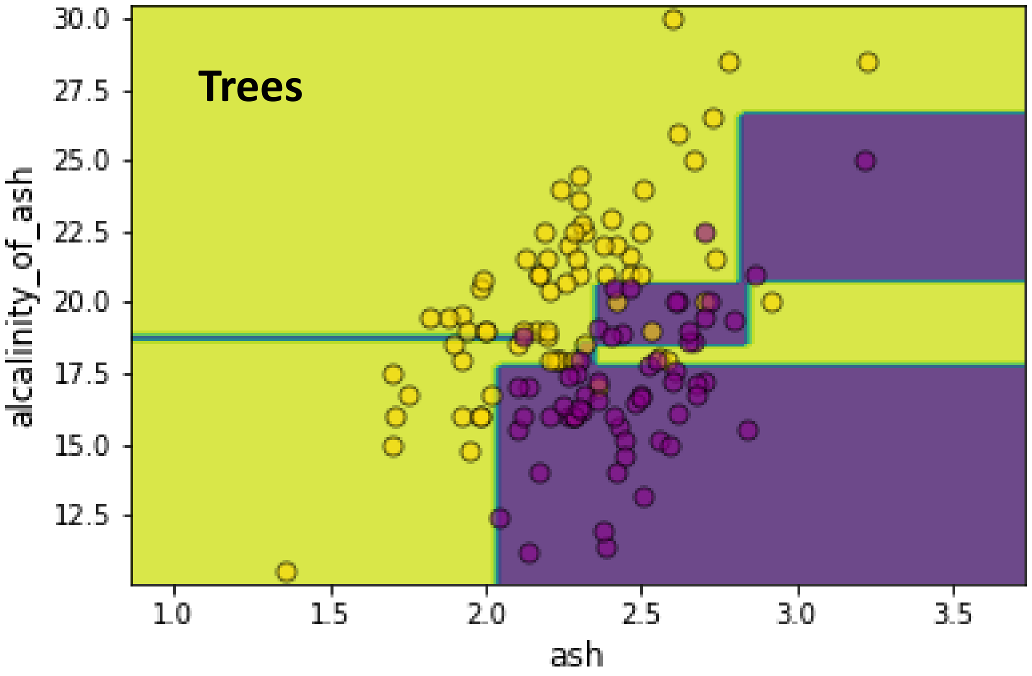
- Any classification model divides the feature space into separate decision regions that can be used to classify unlabeled data points.
- With linear models, a hyperplane is what divides these regions.
- With non-linear models, a more flexible mathematical object is what separates the classes in the feature space.
- This separating object is what is called **decision boundary,** a hypersurface that defines the boundary of each classification region.



Example of non-linear classification models



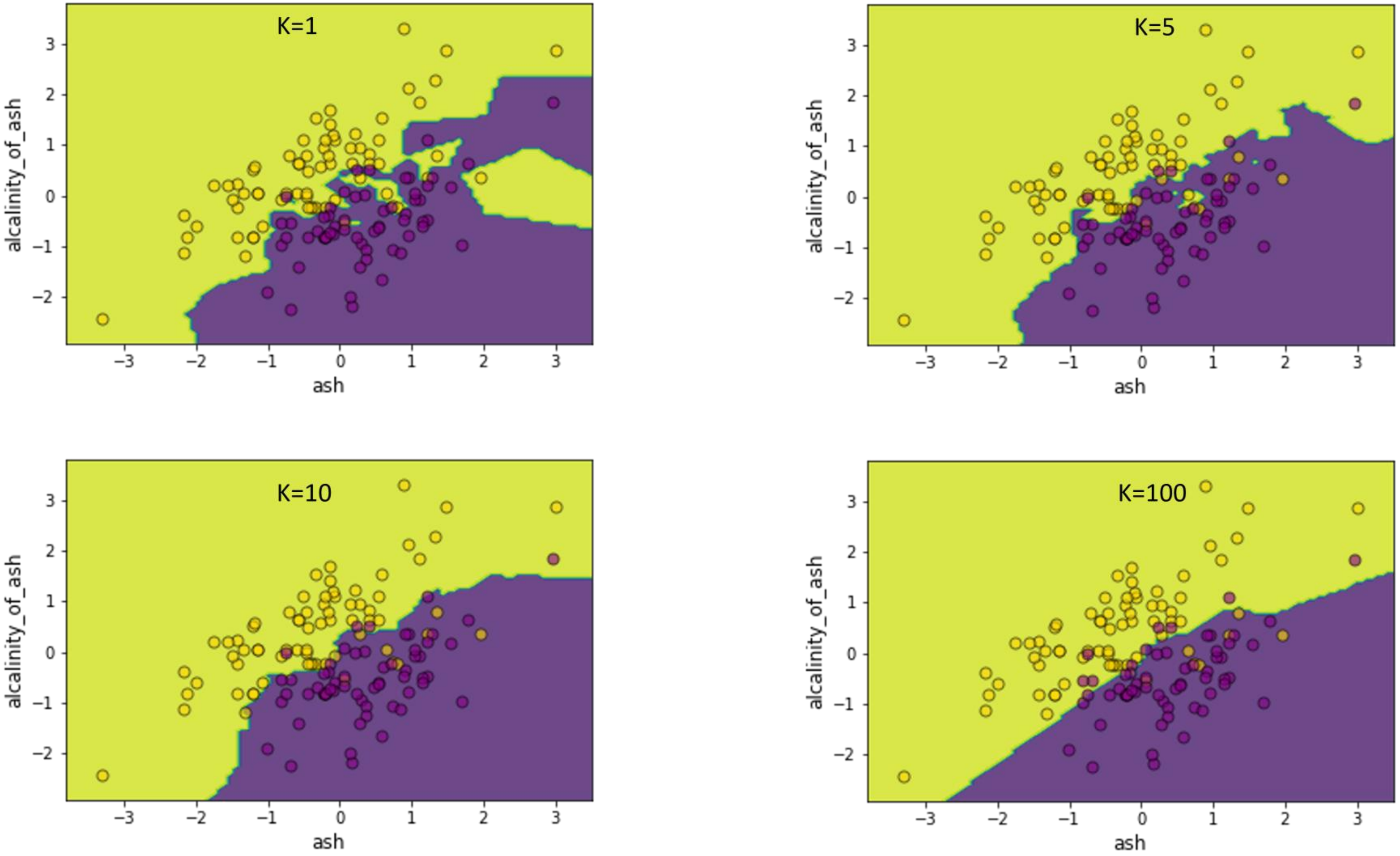




Example of non-linear classification models

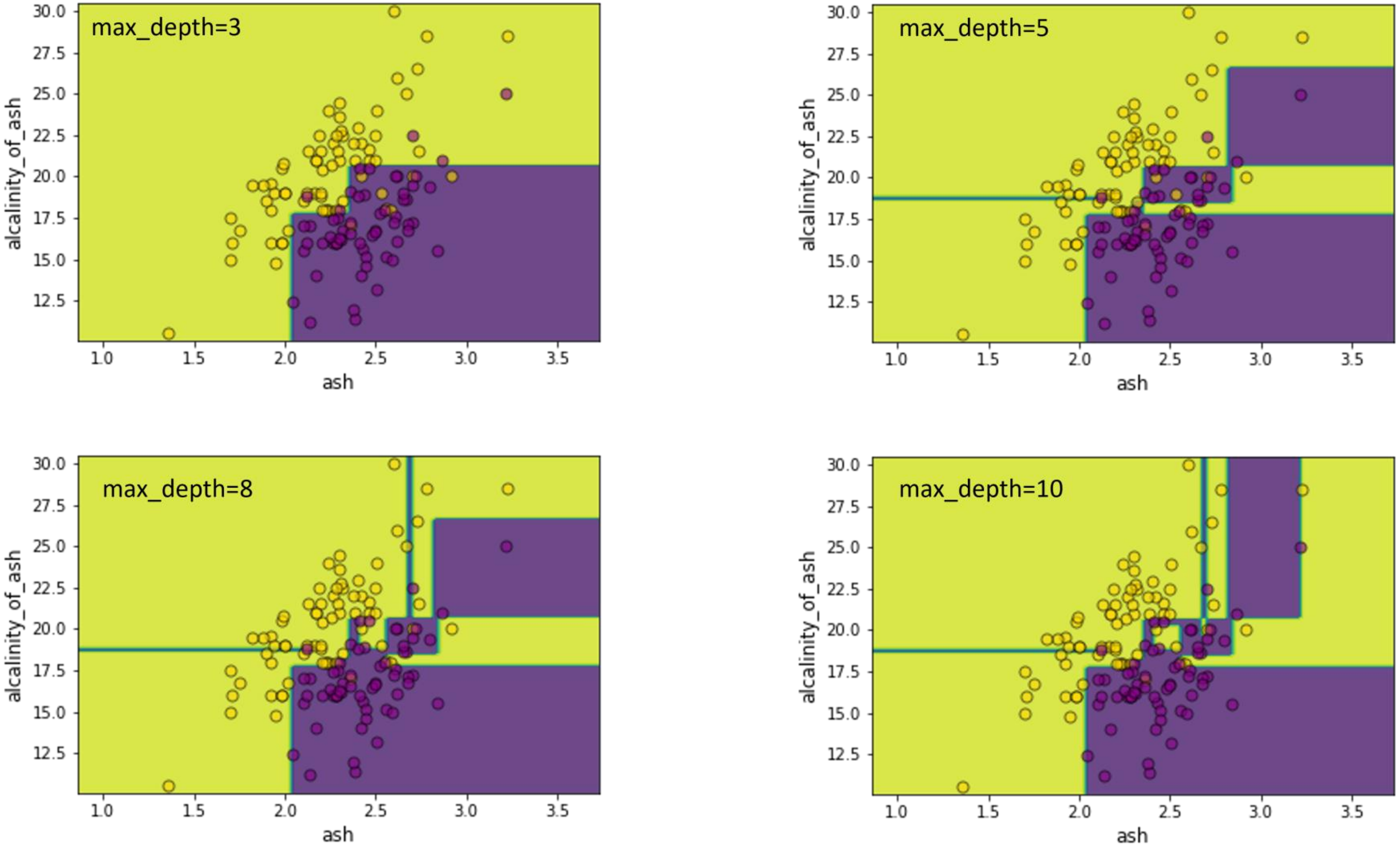
- KNN and Trees have extra settings (hyperparameter) that can be changed to change the complexity of the model itself.

- How does the complexity of KNN change with changing K?



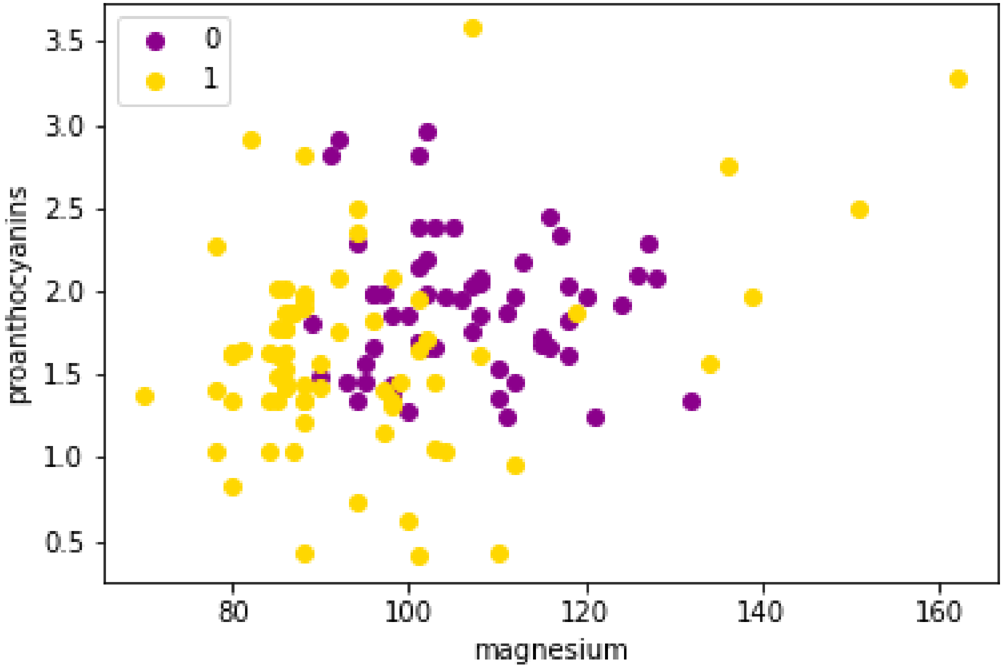
Example of non-linear classification models

- KNN and Trees have extra settings (hyperparameter) that can be changed to change the complexity of the model itself.
- How does the complexity of KNN change with changing K?



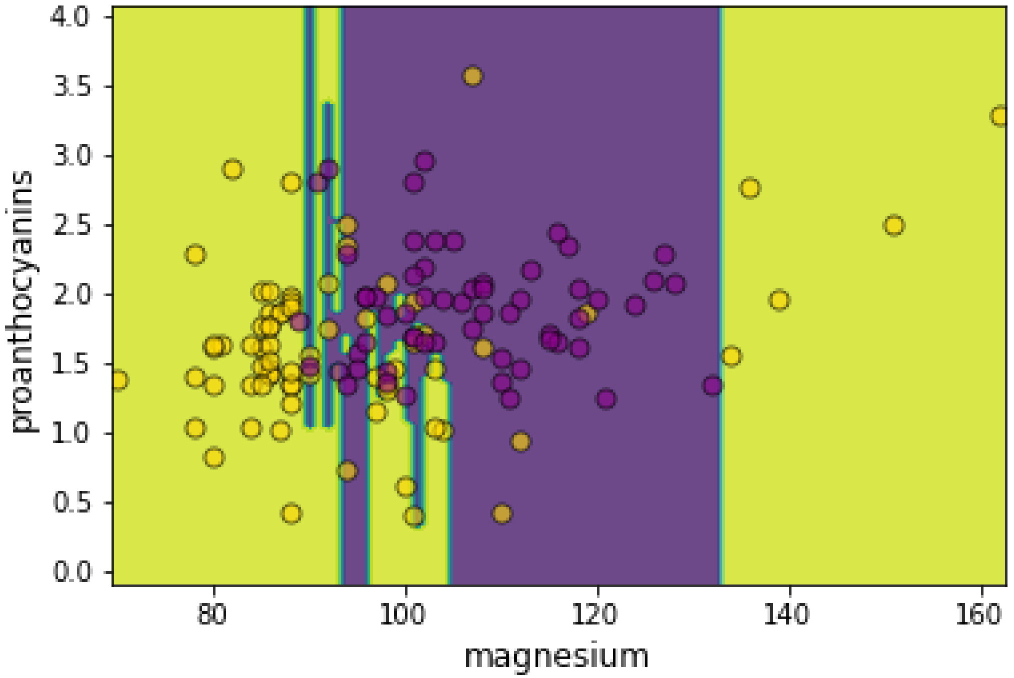
Decision Boundaries & Feature Scaling

- Assume we're interested in predicting the wine type 0 or 1, from the two features: magnesium and proanthocyanins.
- These two features have different range of values.
- Would that affect the decision boundary of KNN or decision tree?

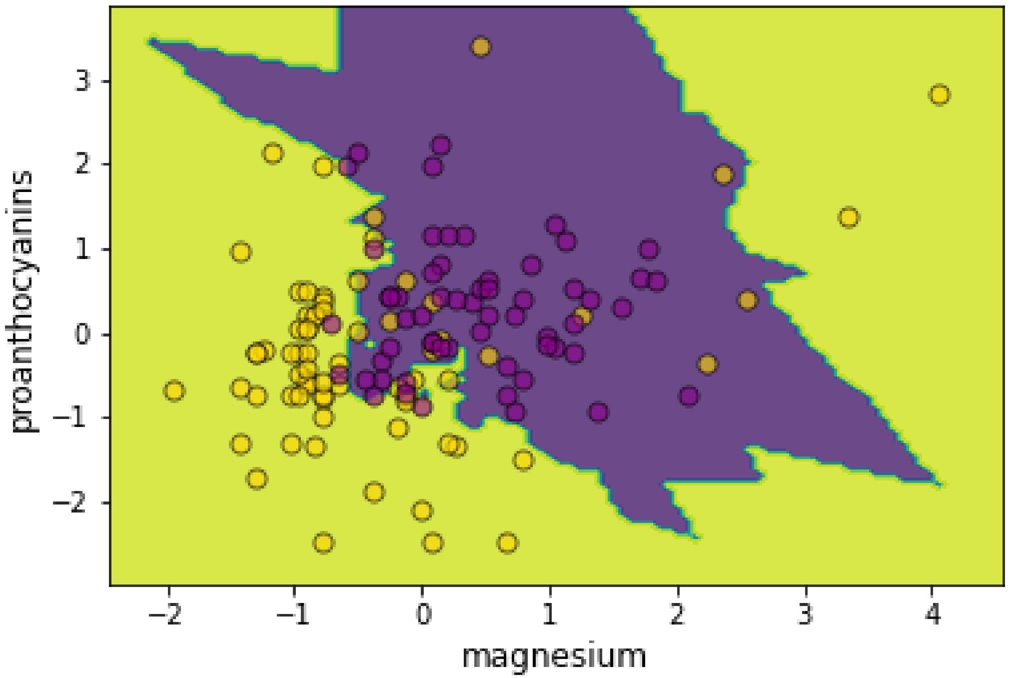


Decision Boundaries & Feature Scaling

KNN – no scaling

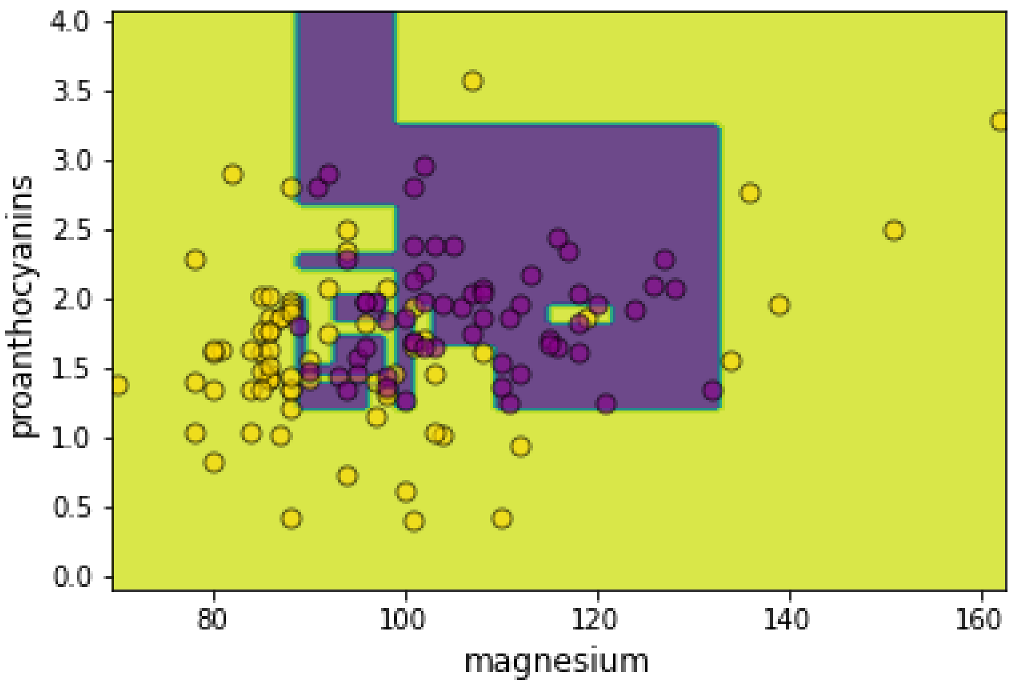


KNN – scaling

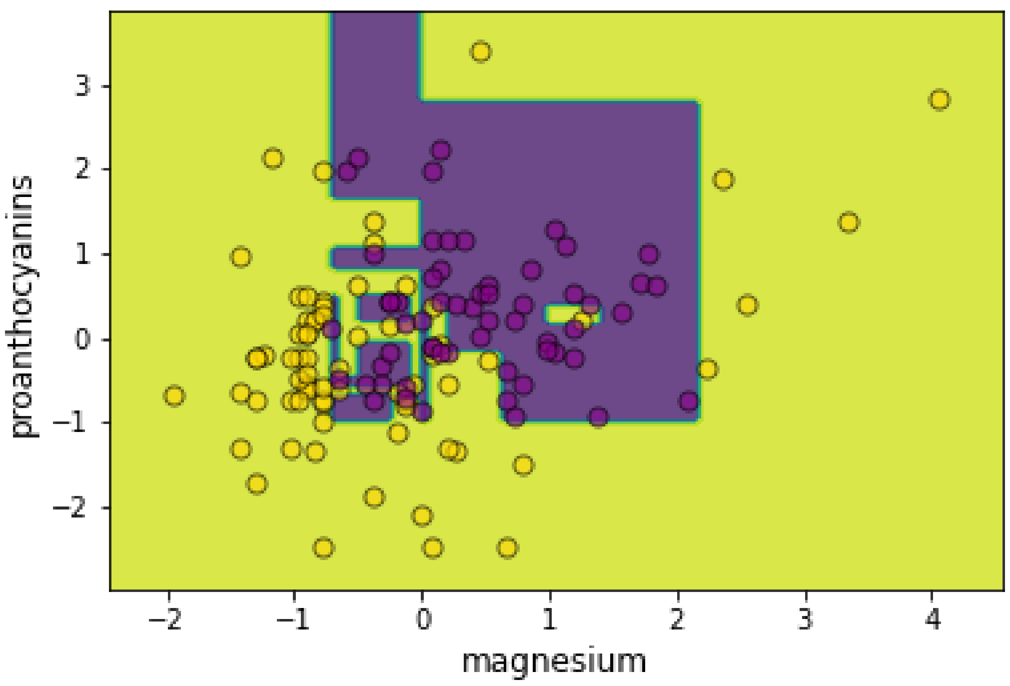


Decision Boundaries & Feature Scaling

Trees – no scaling



Trees – scaling



Linear vs Non-Linear Decision Boundaries

Linear

- Logistic regression (line)
- SVMs (linear kernels)

Non-Linear

- KNN (no compact decision boundary model)
- SVMs with the kernel trick
- Neural Networks
- Decision Trees / Random Forests (half-planes)

Linear vs Non-Linear Models

Linear

- Simpler models
- Less chance of overfitting a model
- Less memory usage: storing trained linear models only requires storing the weights

Non-Linear

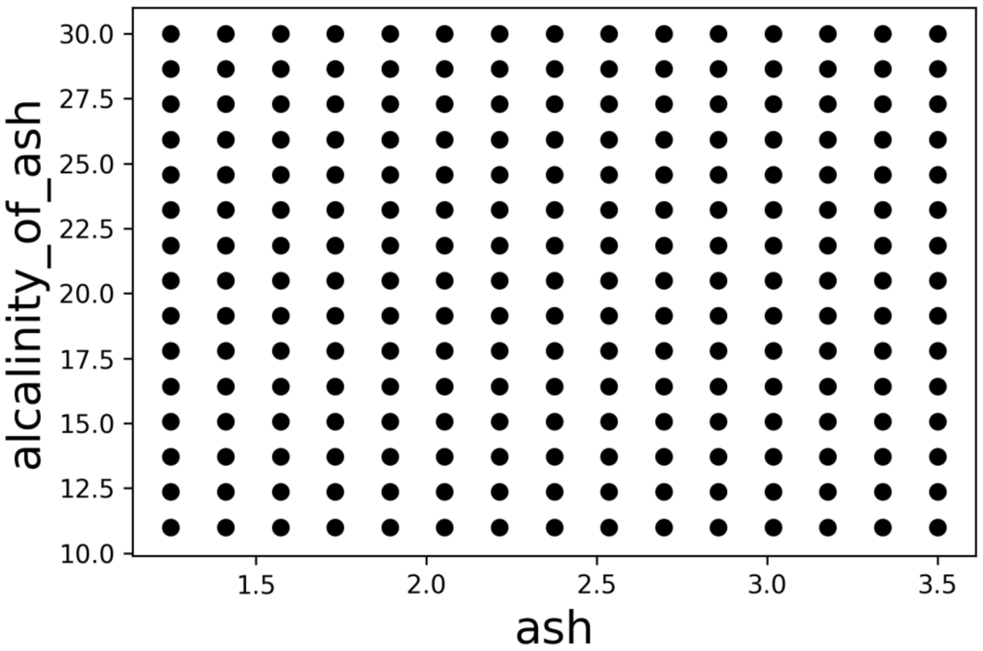
- More complex models
- Higher chance of overfitting a model
- Might require more memory usage

Do we need non-linear models?

- It depends on the data set.
- For high dimensional data (number of features much greater than the number of rows), we might need to stick with simpler models.

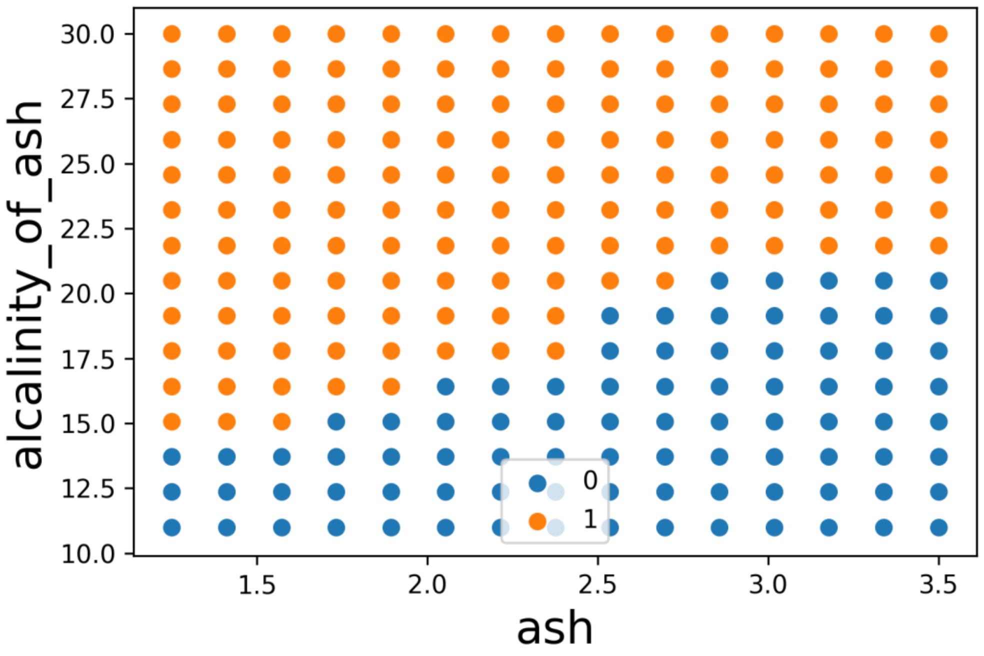
Additional Slides

Visualizing Decision Boundaries Learned by a Model



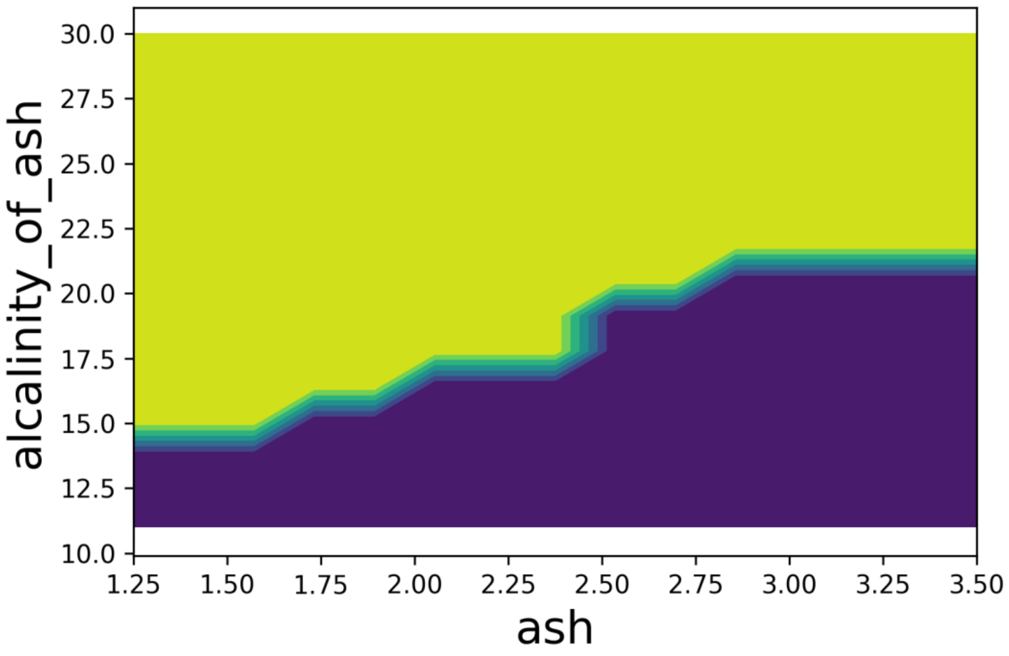
- We define a grid of points over the feature space

Visualizing Decision Boundaries Learned by a Model



- "Query" the model to get a prediction for each grid point

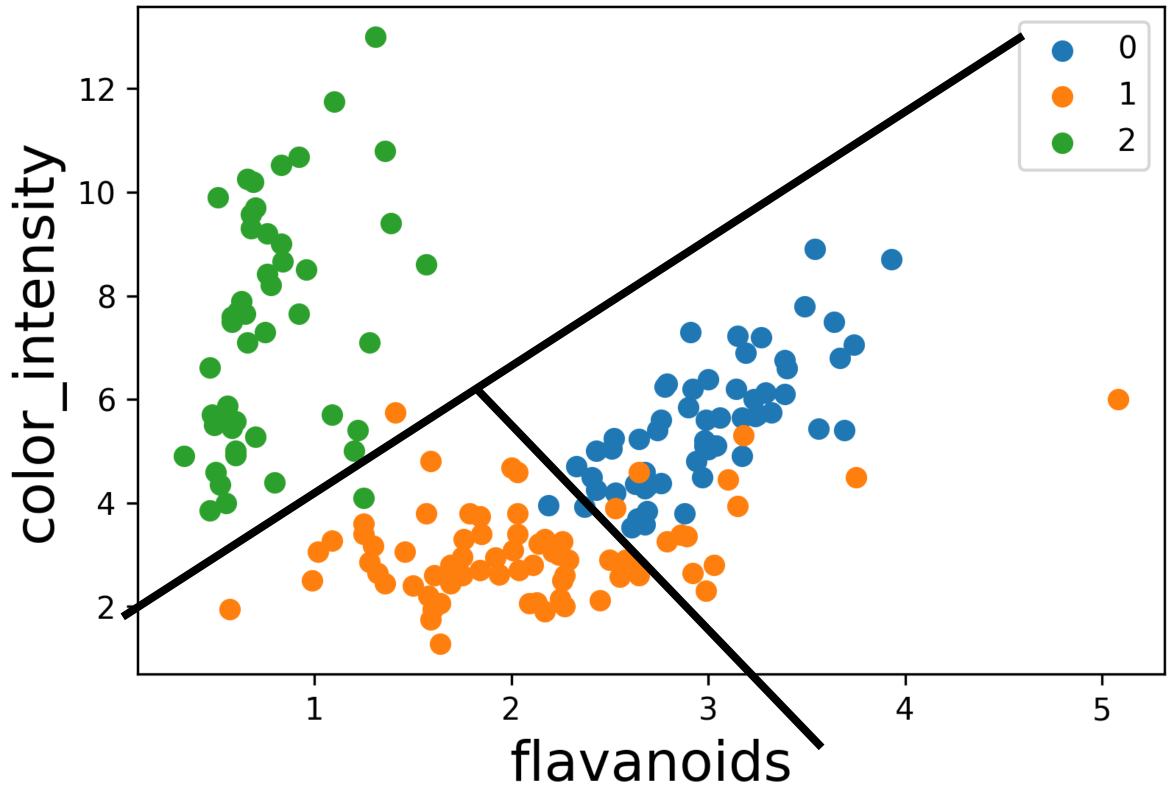
Visualizing Decision Boundaries Learned by a Model



- Plot the grid predictions using a contour plot that interpolates (e.g., averages) between grid points

Decision Boundary Quality Depends on Features and the Model

Wine Examples



Wine Examples