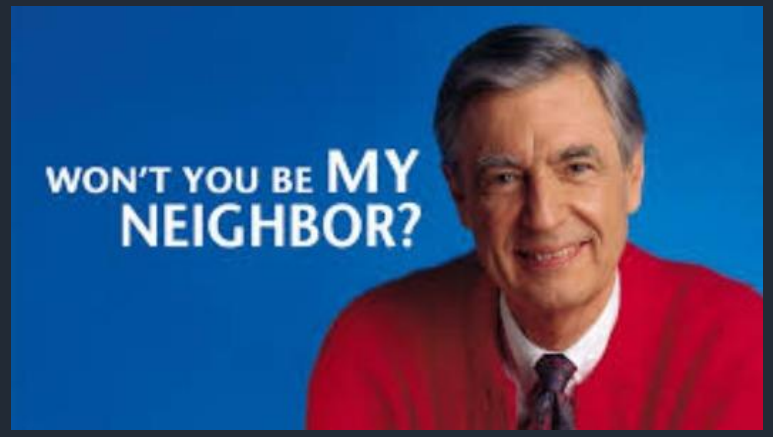
Researching ML Algorithms: K-Nearest Neighbors (Regression)

Connor, Ben H., Lucas, Lindsey



Introduction

- How It Works
- Advantages and Disadvantages
- Similarities and Differences
- Data Processing
- Hyperparameters

- Helps build other algorithms
  - DBSCAN
  - Isomap
- Fun fact: used in Photoshop Brushstrokes

How It Works

- Similar to how k-means works, the algorithm uses a predefined number of training points to assess the class of the new point and predict it based off the class of the nearest points
- Various metrics can be used to calculate distance between points, such as Euclidean distance or Manhattan distance
- The algorithm is very simple, yet has found success in a variety of applications from classifying handwriting to assessing satellite images

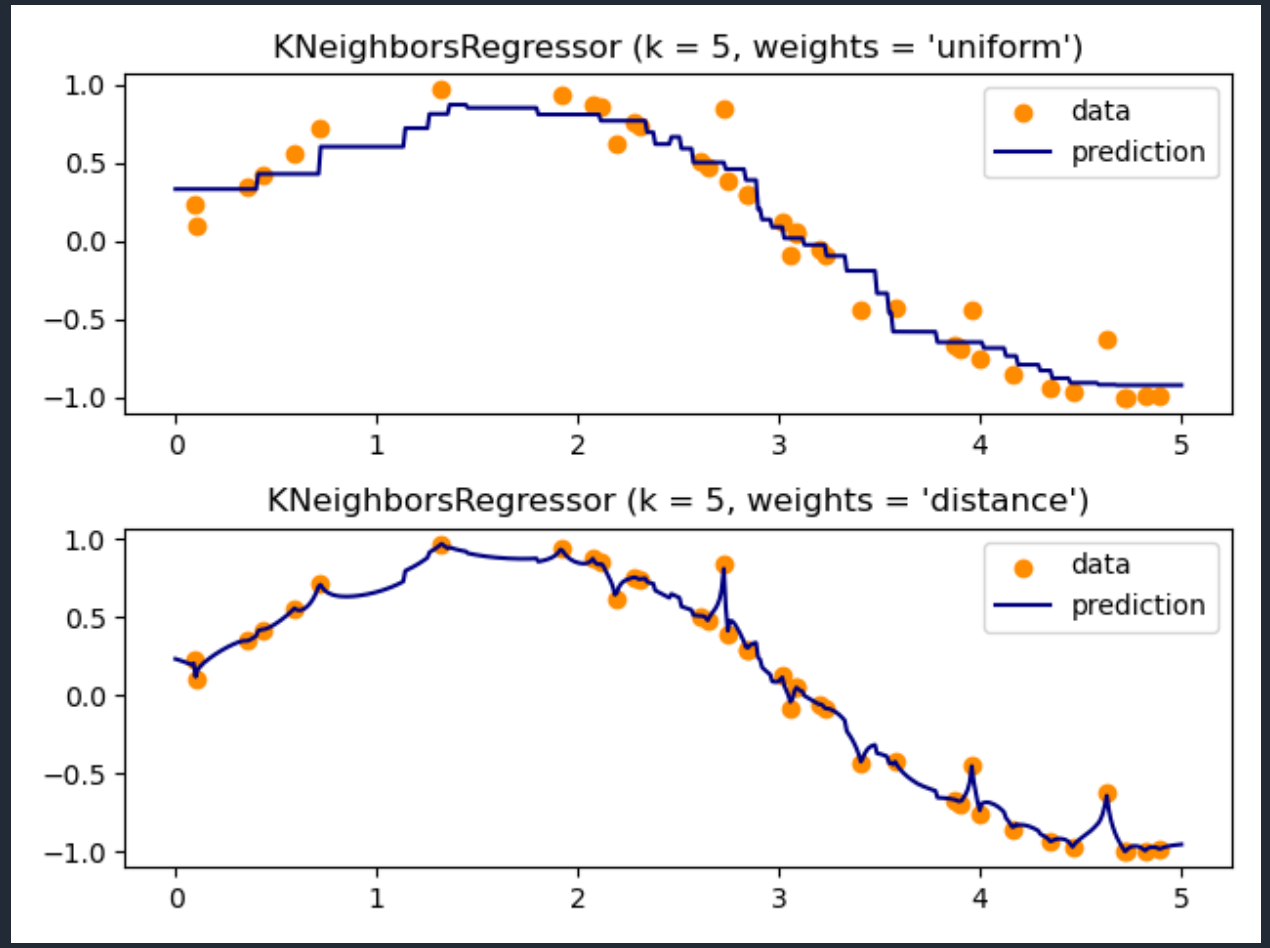
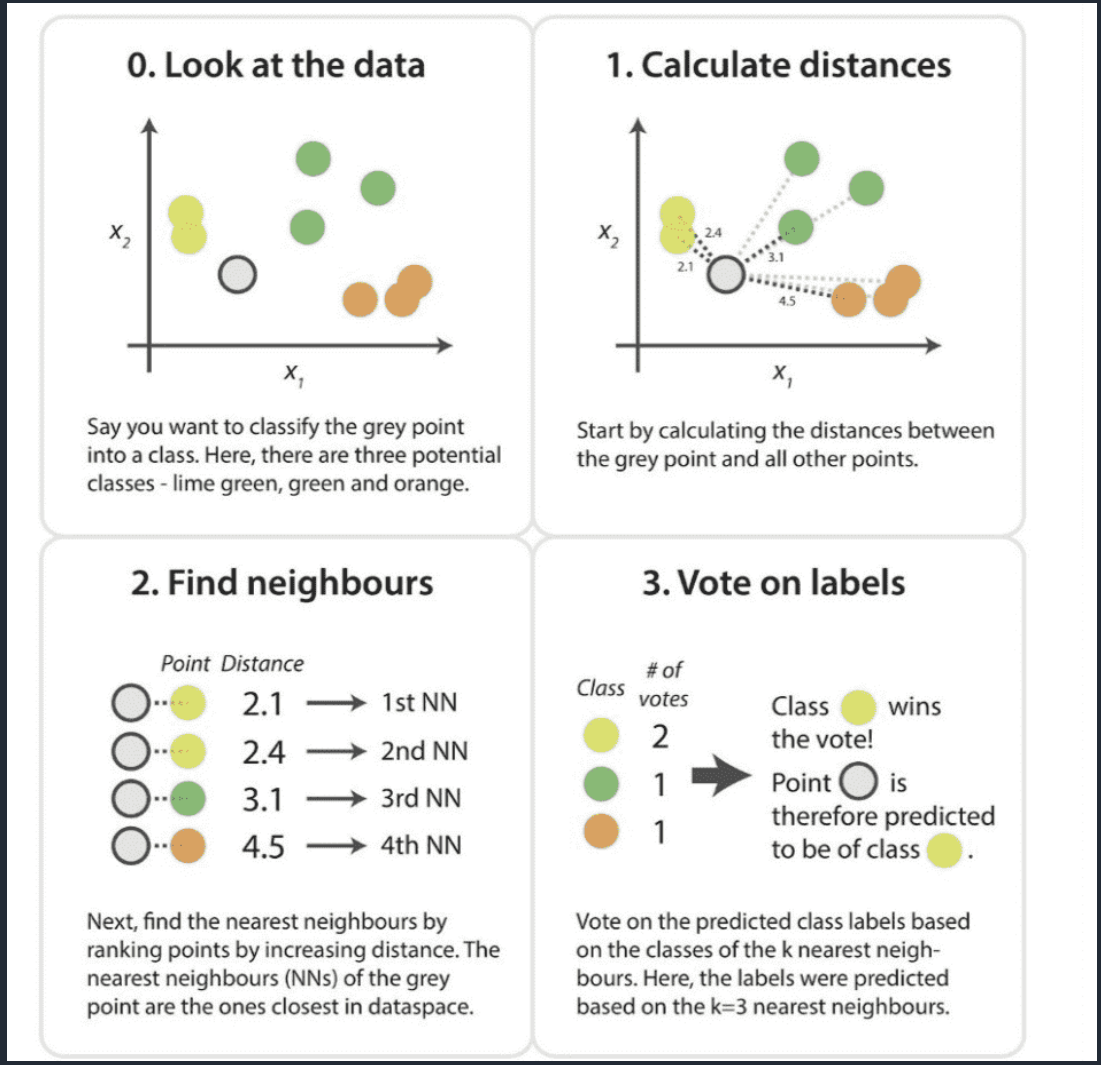


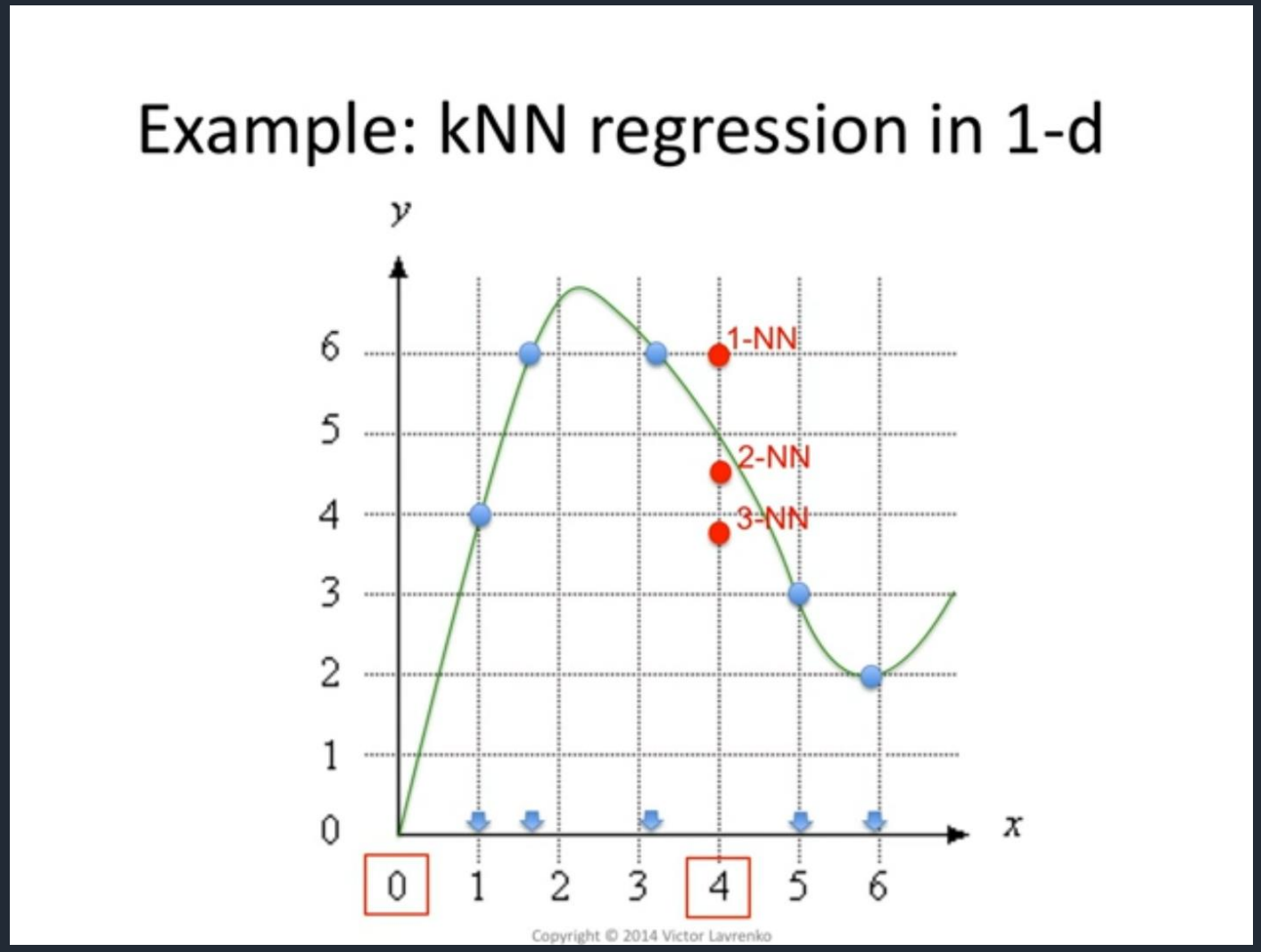
Figure from Source 2

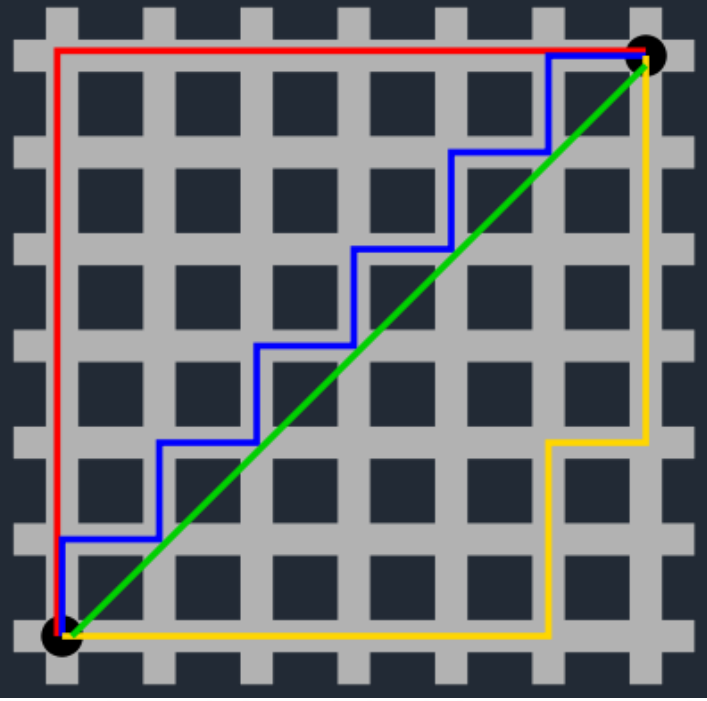
How It Works



Source: https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4

How It Works





Distance functions

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k}|x_i - y_i|$

Figure 1 from Source 7, Figure 2 from Source 11

Advantages & Disadvantages

- Easy to implement
- Does not require a model or tuning of multiple parameters
- Can be used for both classification and regression
- Ability to tolerate outliers by only looking at nearest points

- Increase in independent variables significantly decreases algorithm speed
- Increasing K does not lead to more meaningful data; increasing K returns a value closer and closer to the mean

From Source 4

Similarities & Differences Between….

Linear Regression

- Supervised algorithm
- Independent & dependent variables
- Fits a linear function to the data points in order to make predictions
- Specialized for linear data

K-Nearest Neighbors Regression

- Supervised algorithm
- Independent & dependent variables
- Uses nearest data points to make a prediction of new points, does not use a linear function
- Suitable for non-linear data

Similarities and Differences Between….

K-Means

- Focused on finding clusters
- Unsupervised method
- Relies on distance for grouping points
- Method requires no training, finds clusters through iteration
- Requires scaling to perform effectively

K-Nearest Neighbors Regression

- Focused on regression
- Can be unsupervised, but more likely to find use as supervised
- Similarly relies on distance to make value calculation
- Uses training data to find closest sample of points and learn structure
- Also requires scaling

Data Processing Steps Required

- Extracting independent and dependent variables

  ex: x= data_set.iloc[:, numerical_data_cols].values

  y= data_set.iloc[:, target_col].values

- Splitting the dataset into training and test set.

  ex: x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 0.25, random_state=0)

- Feature scaling(i.e. standardization):

  ex: st_x= StandardScaler()

  x_train= st_x.fit_transform(x_train)

  x_test= st_x.transform(x_test)

Hyperparameters

- Different algorithms of K-Nearest Neighbors have different hyperparameters, but all need to specify or find number of neighbors
- Radius-based KNN requires a radius instead of the number of nearest neighbors, this radius controls how large of an area around a point that others are considered neighbors
- KDTree and BallTree algorithms both use a leaf size parameter which controls the amount of data contained in their partitions
- Weights can be set for various user-defined or built-in functions, like uniform weights or weights based on distance (the further the point, the lower the weight)

Conclusion

- K-nearest neighbors is an essential algorithm in Machine Learning
- For higher speed and/or accuracy, other algorithms might be preferred

From Source 4

1. Appendix (ctrl + click to open link)

1. Regression Example with K-Nearest Neighbors in Python
2. Nearest Neighbors Regression — plot and code example
3. K-Nearest Neighbors with python code
4. Machine Learning Basics with the K-Nearest Neighbors Algorithm
5. sklearn.neighbors.KDTree — Algorithm Examples
6. sklearn.neighbors.BallTree — Algorithm Examples
7. [Video] kNN Regression Example
8. K-Nearest Neighbors in Python + Hyperparameters Tuning
9. K Nearest Neighbors Regression with Python
10. [Video] Nearest Neighbor Classification in Python
11. https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4