

AAIL100: Introduction to Data Science

Ragesh Jaiswal, CSE, IIT Delhi

Classification

- Course content review:
 - Hypothesis testing: *Is the observation due to chance or otherwise?*
 - Parameter Estimation: *Can we estimate some parameter of the population from a sample?*
 - Linear regression: *Can we predict the value of one variable as a linear function of another?*

Classification

- Course content review:
 - Hypothesis testing: *Is the observation due to chance or otherwise?*
 - Parameter Estimation: *Can we estimate some parameter of the population from a sample?*
 - Linear regression: *Can we predict the value of one variable as a linear function of another?*
 - **Classification**: *Given examples with correct **predictions**, can we learn from those examples to make good **predictions** in the future?*
 - Type of prediction: Which **class** does the example belong to?
 - Examples: *Is this online order fraudulent? Does this patient have cancer?* These are examples of **Binary Classification**, where there are only two classes (typically Yes/No).

Classification

- **Classification**: *Given examples with correct **predictions**, can we learn from those examples to make good **predictions** in the future?*
 - Type of prediction: Which **class** does the example belong to?
 - Examples: *Is this online order fraudulent? Does this patient have cancer?* These are examples of **Binary Classification**, where there are only two classes (typically Yes/No).
 - **Definitions**:
 - The individual or situation for which we would like to make a prediction is called an **observation**.
 - Each observation has multiple **attributes**, which are known (e.g., the total value of an online order, a patient's blood report).
 - Each observation has a **class**, which is the answer to the question we care about (e.g., fraudulent or not, cancer or not).
 - We have access to **training data**: a set of observations, each with a known class.
 - The collection of these pre-classified observations is also called a **training set**.
 - A **classification algorithm** is going to analyze the training set, and then come up with a **classifier**: an algorithm for predicting the class of future observations.

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
 - Type of prediction: Which *class* does the example belong to?
 - Examples: Is this online order fraudulent? Does this patient have cancer? These are examples of **Binary Classification**, where there are only two classes (typically Yes/No).
 - **Definitions**:
 - The individual or situation for which we would like to make a prediction is called an **observation**.
 - Each observation has multiple **attributes**, which are known (e.g., the total value of an online order, a patient's blood report).
 - Each observation has a **class**, which is the answer to the question we care about (e.g., fraudulent or not, cancer or not).
 - We have access to **training data**: a set of observations, each with a known class.
 - The collection of these pre-classified observations is also called a **training set**.
 - A **classification algorithm** is going to analyze the training set, and then come up with a **classifier**: an algorithm for predicting the class of future observations.

• *Classifiers do not need to be perfect to be useful. They can be useful even if their accuracy is less than 100%.*

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with ***Binary classification***.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*

Classification

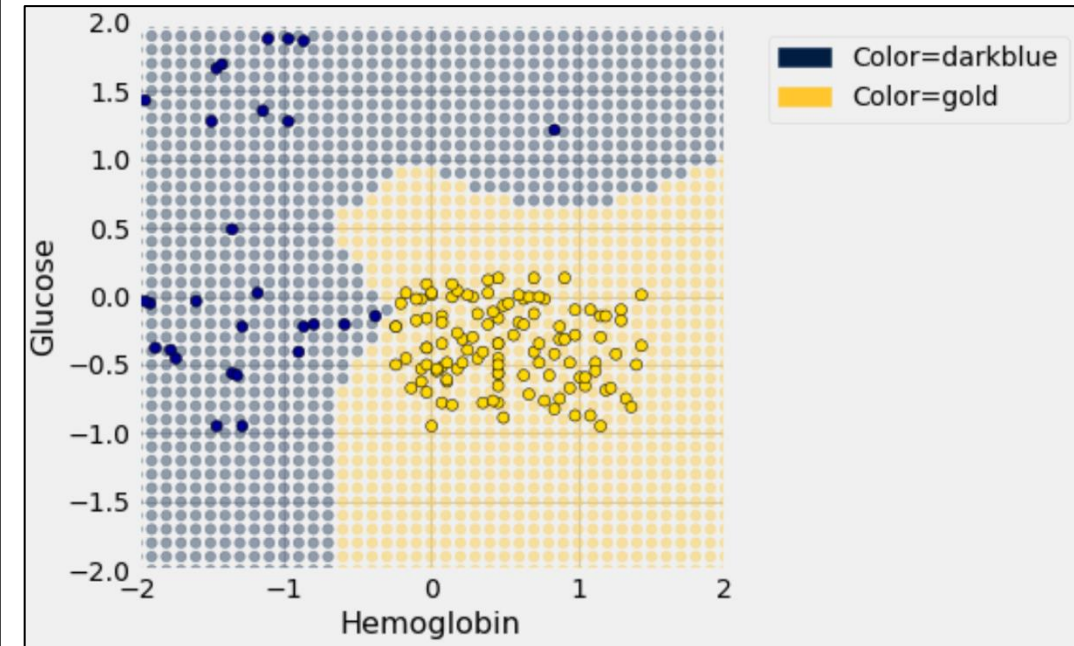
- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with ***Binary classification***.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with ***Binary classification***.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find the point p in the training set that is nearest to the new point and predict its class to be the same as that of p .
 - The above idea implicitly defines a ***decision boundary*** for making predictions on new points.

Classification

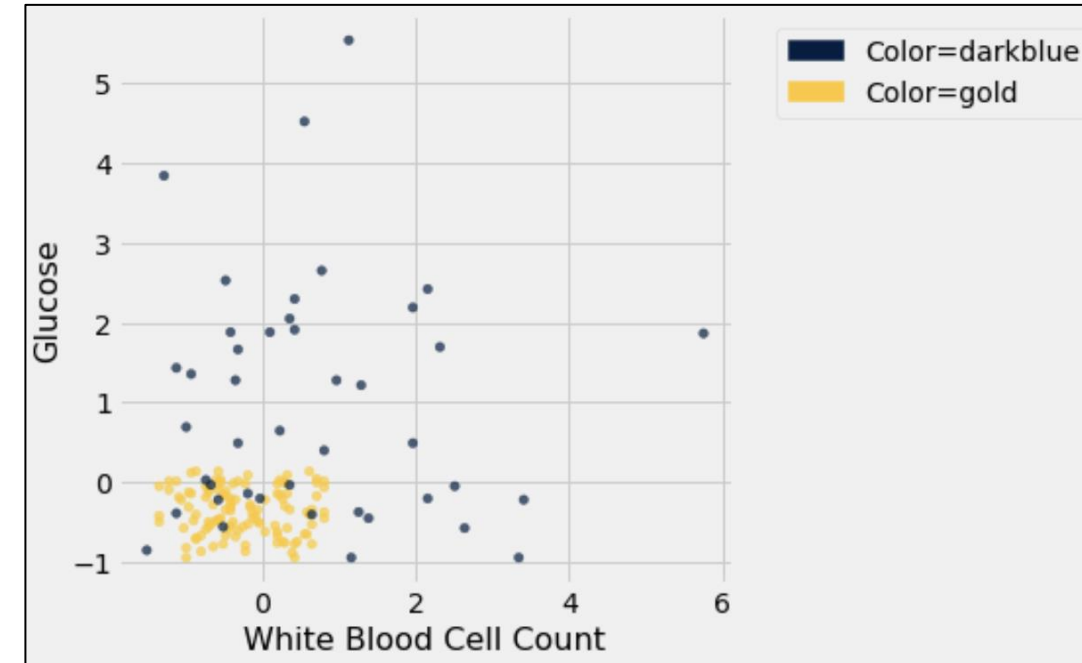
- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find the point p in the training set that is nearest to the new point and predict its class to be the same as that of p .
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.



Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find the point p in the training set that is nearest to the new point and predict its class to be the same as that of p .
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

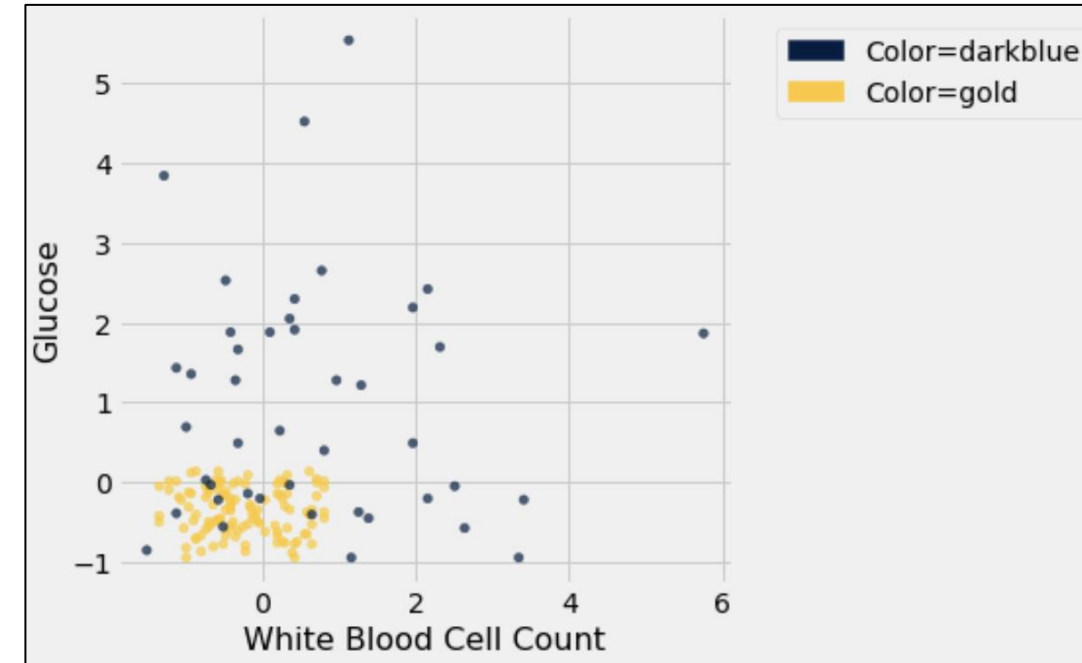
- Sometimes, the decision boundary is not clean



Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find the point p in the training set that is nearest to the new point and predict its class to be the same as that of p .
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

- Sometimes, the decision boundary is not clean

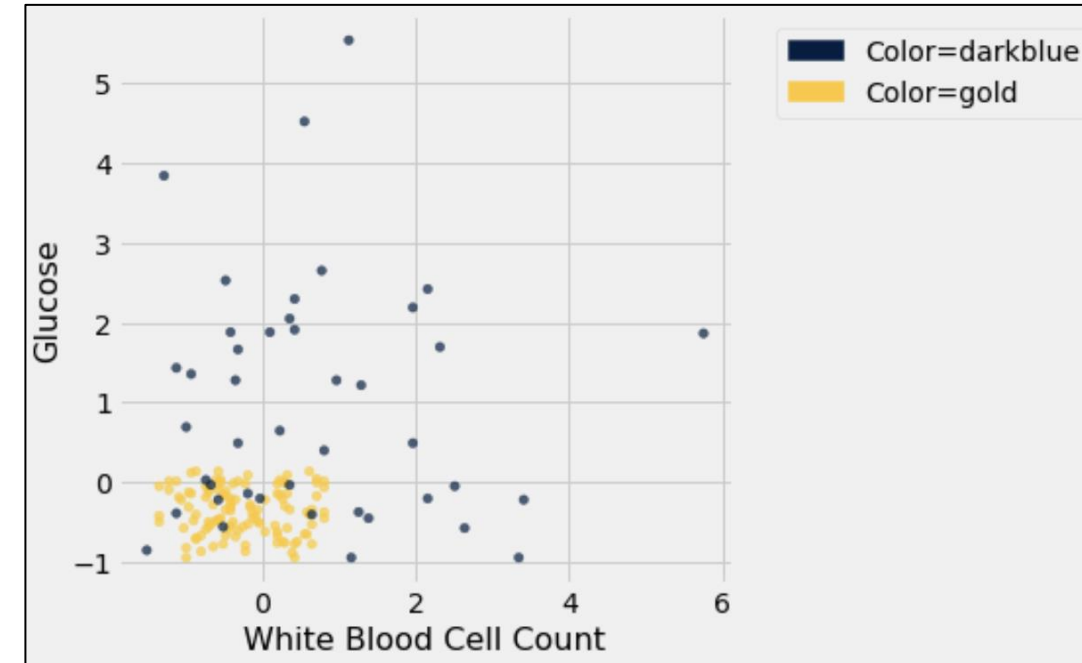


Can you suggest a classification scheme?

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find the point p in the training set that is nearest to the new point and predict its class to be the same as that of p .
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

- Sometimes, the decision boundary is not clean



- Idea: Find k nearest neighbors (KNN) and classify based on the majority vote.

Classification

- **Classification**: *Given examples with correct **predictions**, can we learn from those examples to make good **predictions** in the future?*
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the **similarity** of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: **Find k nearest neighbors (KNN) and classify based on the majority vote.**
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

- How do we evaluate the classifier?
 - Issue: We can't use the training set to check.

Classification

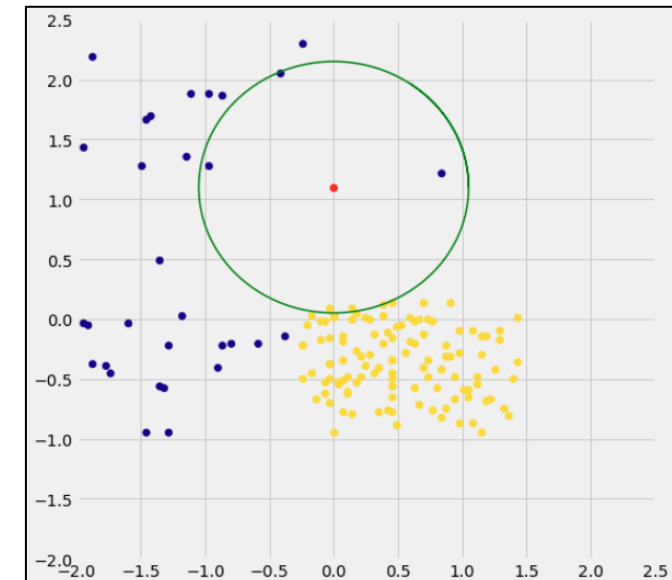
- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: *Find k nearest neighbors (KNN) and classify based on the majority vote.*
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

- How do we evaluate the classifier?
 - Issue: We can't use the training set to check.
 - Idea: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.
 - The hope is that our classifier will perform about as well on the underlying population as it did on our test set.

Classification

- **Classification**: Given examples with *correct predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - What is a good measure of similarity?
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: *Find k nearest neighbors (KNN) and classify based on the majority vote.*
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

- How do we evaluate the classifier?
 - Issue: We can't use the training set to check.
 - Idea: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.
 - The hope is that our classifier will perform about as well on the underlying population as it did on our test set.

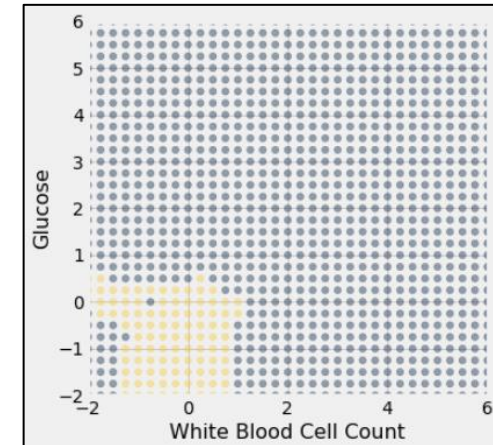


KNN with $k=5$. What is the predicted label for the red point?

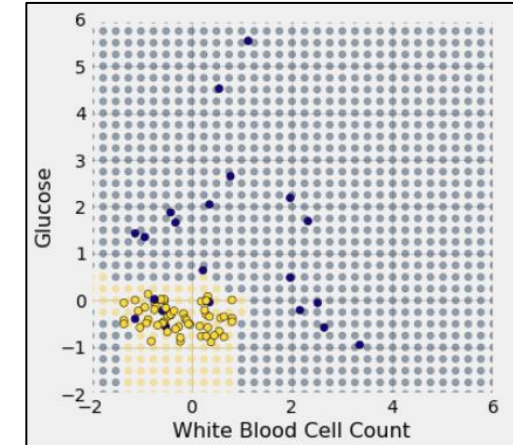
Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - What is a good measure of similarity?
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: *Find k nearest neighbors (KNN) and classify based on the majority vote.*
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.

- How do we evaluate the classifier?
 - Issue: We can't use the training set to check.
 - Idea: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.
 - The hope is that our classifier will perform about as well on the underlying population as it did on our test set.



Decision boundary
based on the training set



Performance on the test
set

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - What is a good measure of similarity?
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find *k nearest neighbors (KNN)* and classify based on the majority vote.
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.
 - Evaluation: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.

• Multiple attributes:

- Each attribute can yield new information.
- So, running KNN with multiple attributes may improve the classifier.

WaveletVar	WaveletSkew	WaveletCurt	Entropy	Class
3.6216	8.6661	-2.8073	-0.44699	0
4.5459	8.1674	-2.4586	-1.4621	0
3.866	-2.6383	1.9242	0.10645	0
3.4566	9.5228	-4.0112	-3.5944	0
0.32924	-4.4552	4.5718	-0.9888	0
4.3684	9.6718	-3.9606	-3.1625	0
3.5912	3.0129	0.72888	0.56421	0
2.0922	-6.81	8.4636	-0.60216	0
3.2032	5.7588	-0.75345	-0.61251	0
1.5356	9.1772	-2.2718	-0.73535	0

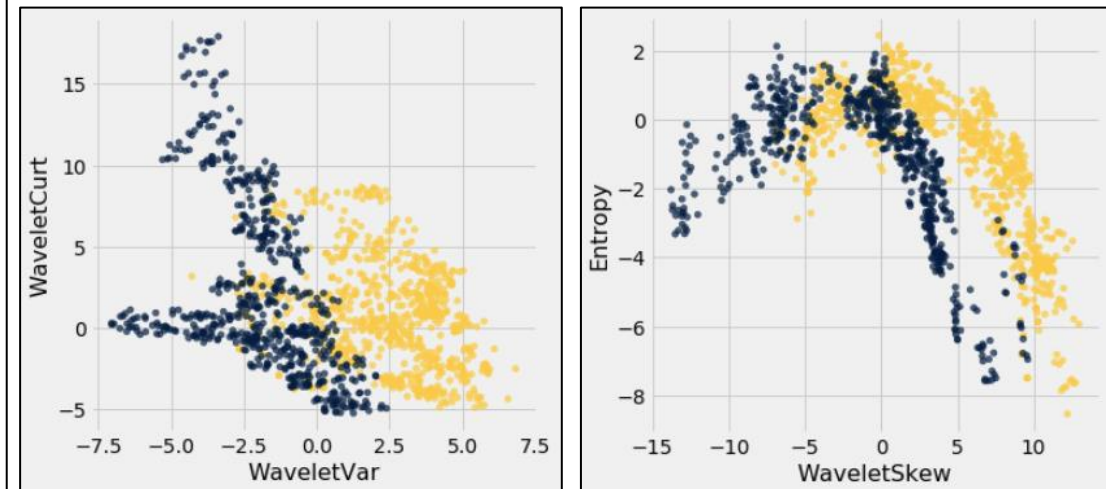
Counterfeit banknotes data

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - What is a good measure of similarity?
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find *k nearest neighbors (KNN)* and classify based on the majority vote.
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.
 - Evaluation: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.

- **Multiple attributes**:

- Each attribute can yield new information.
- So, running KNN with multiple attributes may improve the classifier.



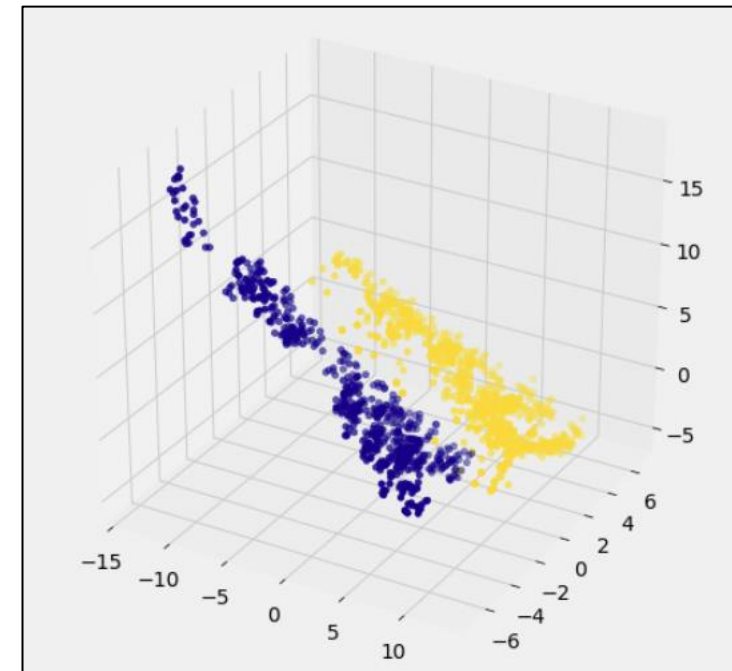
Scatter plots with two attributes

Classification

- **Classification**: Given examples with correct *predictions*, can we learn from those examples to make good *predictions* in the future?
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the *similarity* of a new observation (i.e., with unknown class) with the known observations in the training set.
 - What is a good measure of similarity?
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: Find *k nearest neighbors (KNN)* and classify based on the majority vote.
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.
 - Evaluation: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.

- **Multiple attributes**:

- Each attribute can yield new information.
- So, running KNN with multiple attributes may improve the classifier.



Scatter plot with three attributes

Classification

- **Classification**: *Given examples with correct **predictions**, can we learn from those examples to make good **predictions** in the future?*
- Let's start with **Binary classification**.
- The simplest binary classifier is based on the **similarity** of a new observation (i.e., with unknown class) with the known observations in the training set.
 - *What is a good measure of similarity?*
 - For numerical attributes: Euclidean distance after conversion to standard units.
 - Classification idea: **Find k nearest neighbors (KNN) and classify based on the majority vote.**
 - The above idea implicitly defines a **decision boundary** for making predictions on new points.
 - Evaluation: Randomly split the labeled data into a **test set** and a **training set**. Use the test set to evaluate the classifier built using the training set.

- **Multiple attributes**:
 - *Each attribute can yield new information.*
 - *So, running KNN with multiple attributes may improve the classifier.*

End