ECON484 Machine Learning
2. First kNN Model

Lecturer:
Bora GÜNGÖREN
bora.gungoren@atilim.edu.tr

K-Nearest Neighbor (kNN)

- kNN is a very simple machine learning technique that is used for classification (and regression).

  - It is a **supervised learning technique** because it requires an initial set of classes to be identified.

  - The initial set of classes is presented as a set of labeled data points, where the labels are the classes. This is the required training set.

  - Then each new data point presented is compared with the training set and labeled.

  - You could add new data points to the training set or not. That's up to your implementation.

K-Nearest Neighbor (kNN)

- The method is simple.

- For each new data point

  - We calculate the distances between the data points in the training data set and the new data point.

    - **How we define the "distance" concept is very important.**

  - We select the k nearest neighbors, so that we have a sample showing the classification already existing in the neighborhood.

K-Nearest Neighbor (kNN)

- The method is simple.

- For each new data point

  - We count the classification assignments, and find the majority, and assign the new data point to this majority. ("You belong to the majority in the neighborhood")

    - The counting process is often referred to as **a vote**.

    - The **rule of majority** may change based on your implementation.

  - Note that this assignment is a **prediction**. So kNN uses a type of regression to predict the classification result.
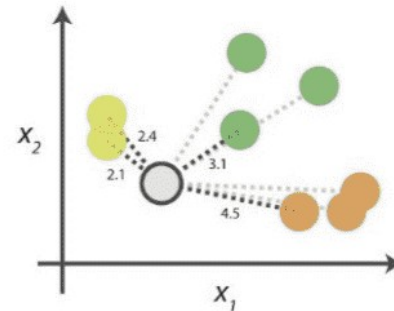
K-Nearest Neighbor (kNN)

0. Look at the data
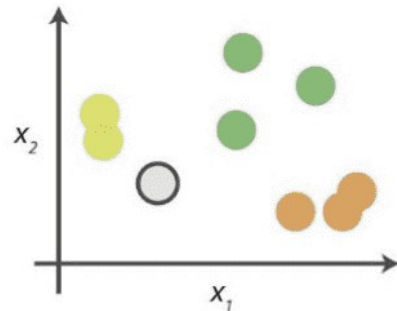
Say you want to classify the grey point into a class. Here, there are three potential classes - lime green, green and orange.

1. Calculate distances

Start by calculating the distances between the grey point and all other points.

2. Find neighbours

Point	Distance	
○ ⋯ ●	2.1	→ 1st NN
○ ⋯ ●	2.4	→ 2nd NN
○ ⋯ ●	3.1	→ 3rd NN
○ ⋯ ●	4.5	→ 4th NN

Next, find the nearest neighbours by ranking points by increasing distance. The nearest neighbours (NNs) of the grey point are the ones closest in dataspace.

3. Vote on labels

Class	# of votes
●	2
●	1
●	1

Class ● wins the vote!

Point ○ is therefore predicted to be of class ●.

Vote on the predicted class labels based on the classes of the k nearest neighbours. Here, the labels were predicted based on the k=3 nearest neighbours.

K-Nearest Neighbor (kNN)
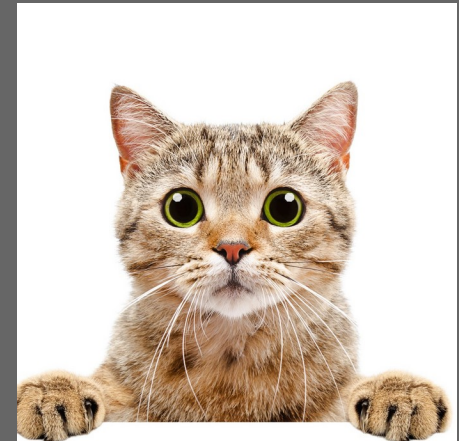
K-Nearest Neighbor (kNN)
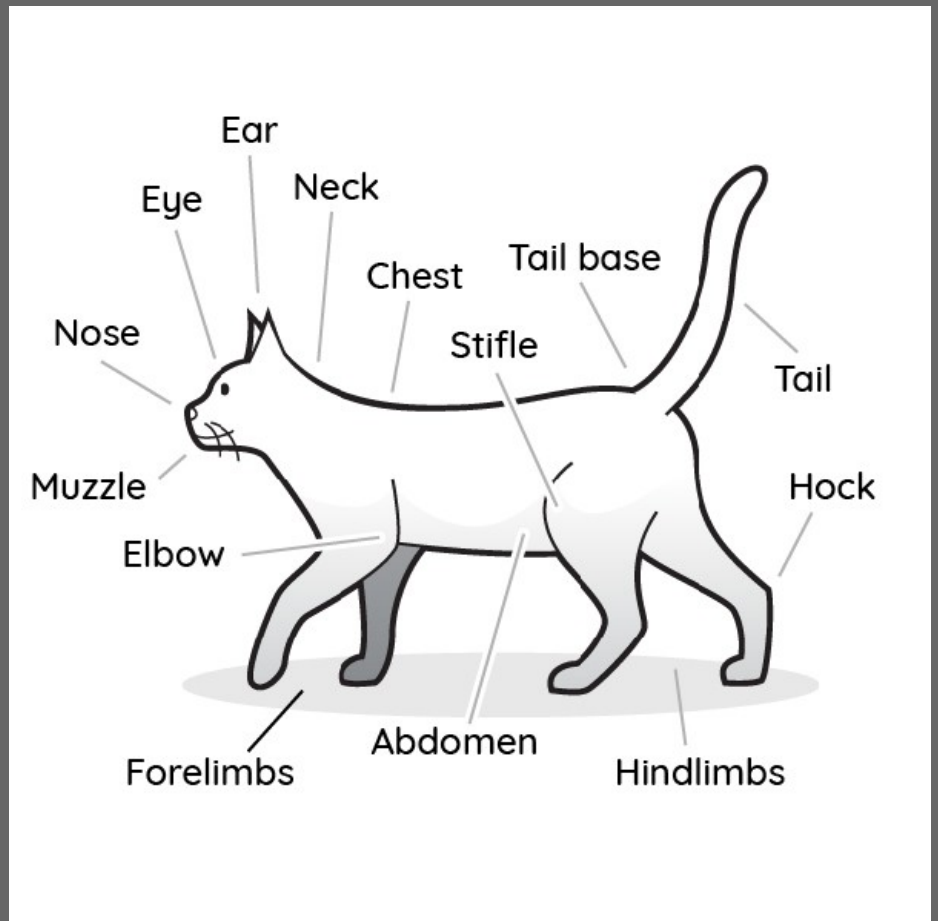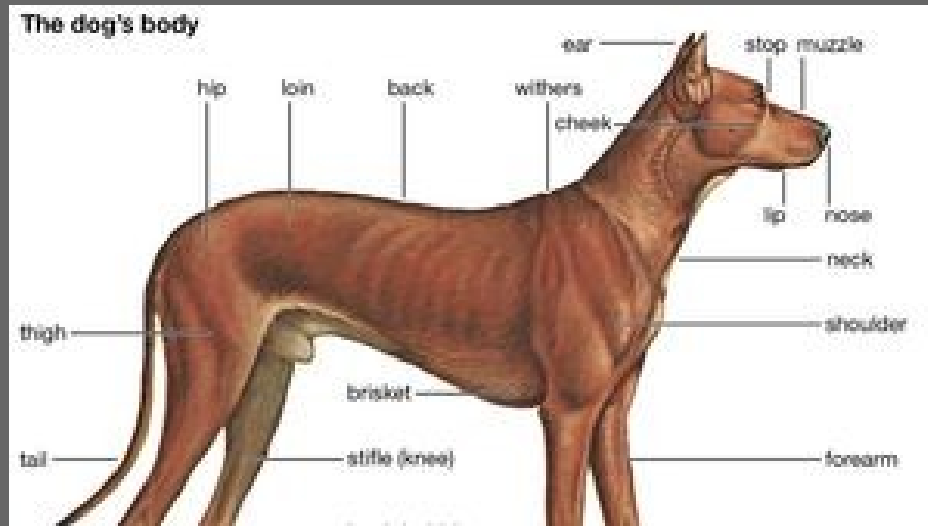
K-Nearest Neighbor (kNN)

- A naive application of kNN without **carefully selected features** and a well designed distance metric usually has a large **misclassification rate**.

  - Misclassification rate is a performance metric that tells you the fraction of the predictions that were wrong, without distinguishing between positive and negative predictions.

  - On the other hand, the misclassification rate can be a very misleading metric when the data set is unbalanced (when the prevalence is either very high or very low).

K-Nearest Neighbor (kNN)

- Let's recall confusion matrix.

    - **Accuracy:** Overall, how often is the classifier correct?

    - **Misclassification Rate:** Overall, how often is it wrong?

    - **Prevalence:** How often does the yes condition actually occur in our sample?

- What happens when we start with a very unbalanced training data set?

    - How can we make sure the training data set represents reality good enough?

K-Nearest Neighbor (kNN)

- kNN is very popular in teaching because it **easily demonstrates** both strengths and weaknesses of supervised learning.

  - Therefore almost all methods compare their performance metrics with kNN.

  - If some method performs worse than kNN, it is considered unsuitable.

- How about speed and memory?

  - Given **an easy to compute distance metric** and **a fast search method to identify the nearest neighbors**, kNN runs relatively fast.

  - KNN **tends to use a lot of memory** unless you have advanced data structures to optimize memory use.

  - For some type of data sets, these two are hard to come by, hence kNN is a very bad choice.

- When you are just toying with data sets using a PC, do not use kNN on large data sets unless you are required to do so.

K-Nearest Neighbor (kNN)

- Eager vs Lazy Learners

  - Eager learners spend more time learning and less time predicting.

  - Lazy learners spend less time learning and more time predicting.

  - Which one is kNN? Why?

- Curse of Dimensionality

  - KNN performs better with a lower number of features than a large number of features.

  - To **avoid** overfitting, the needed data will need to grow exponentially as you increase the number of dimensions.

  - Feature selection is important for kNN.

K-Nearest Neighbor (kNN)

- How do we select k?

  - This is a hard choice.

  - We usually try out from k=2 to some upper limit and compare performance metrics for different k values.

  - Why not k=1? Would that make sense in any practical applications?

K-Nearest Neighbor (kNN)

- How do we process data for kNN?

  - Our example is about predicting whether or not someone likes chocolate banana milkshake.

- Here is the training data set.

  - Some variables are categorical.

  - kNN requires a distance and common distance metrics work with numerical data.

  - We are lucky because **most categories are binary**.

  - We could also work with Likert-like scales.

No	Age	Sex	Occupation	Likes Bananas	Likes Chocolate	Lactose Intolerant	Likes Chocolate Banana Milkshake
1	35	Male	Engineer	Yes	Yes	No	Yes
2	36	Male	Teacher	Yes	No	No	No
3	35	Male	Engineer	No	No	No	No
4	39	Male	Medical Docto	No	Yes	Yes	No
5	37	Male	Truck Driver	Yes	Yes	Yes	Yes
6	40	Male	Truck Driver	Yes	Yes	No	Yes
7	24	Male	Teacher	Yes	No	No	Yes
8	58	Male	Chef	Yes	Yes	No	Yes
9	18	Male	Cashier	No	Yes	No	Yes
10	19	Male	Student	Yes	Yes	No	Yes
11	50	Male	Engineer	Yes	Yes	No	Yes
12	43	Male	Teacher	Yes	No	Yes	No
13	59	Male	Retired	Yes	No	No	No
14	44	Male	Waiter	Yes	Yes	Yes	Yes
15	18	Male	Student	Yes	Yes	Yes	No
16	58	Male	Lawyer	Yes	No	Yes	No
17	33	Female	Medical Docto	Yes	Yes	No	Yes
18	65	Female	Retired	No	Yes	No	Yes
19	24	Female	Student	No	Yes	Yes	Yes
20	61	Female	Retired	Yes	Yes	No	Yes
21	23	Female	Student	Yes	Yes	Yes	Yes
22	59	Female	Medical Docto	No	No	Yes	No
23	56	Female	Teacher	Yes	Yes	Yes	No
24	26	Female	Student	No	Yes	No	Yes
25	34	Female	Teacher	No	Yes	No	Yes
26	48	Female	Teacher	Yes	No	No	Yes
27	49	Female	Medical Docto	No	Yes	No	Yes
28	18	Female	Student	Yes	Yes	Yes	Yes
29	44	Female	Engineer	Yes	Yes	No	Yes
30	32	Female	Medical Docto	Yes	Yes	Yes	No
31	20	Female	Student	Yes	Yes	Yes	Yes
32	50	Female	Nurse	No	Yes	Yes	No
33	38	Female	Lawyer	Yes	Yes	No	Yes
34	58	Female	Retired	No	Yes	Yes	No

K-Nearest Neighbor (kNN)

No	Age	Sex	Occupation	Likes Bananas	Likes Chocolate	Lactose Intolerant	Likes Chocolate Banana Milkshake	Binary Sex	Binary Banana	Binary Chocolate	Binary Lactose	Binary Milkshake
1	35	Male	Engineer	Yes	Yes	No	Yes	0	1	1	0	1
2	36	Male	Teacher	Yes	No	No	No	0	1	0	0	0
3	35	Male	Engineer	No	No	No	No	0	0	0	0	0
4	39	Male	Medical Docto	No	Yes	Yes	No	0	0	1	1	0
5	37	Male	Truck Driver	Yes	Yes	Yes	Yes	0	1	1	1	1
6	40	Male	Truck Driver	Yes	Yes	No	Yes	0	1	1	0	1
7	24	Male	Teacher	Yes	No	No	Yes	0	1	0	0	1
8	58	Male	Chef	Yes	Yes	No	Yes	0	1	1	0	1
9	18	Male	Cashier	No	Yes	No	Yes	0	0	1	0	1
10	19	Male	Student	Yes	Yes	No	Yes	0	1	1	0	1
11	50	Male	Engineer	Yes	Yes	No	Yes	0	1	1	0	1
12	43	Male	Teacher	Yes	No	Yes	No	0	1	0	1	0
13	59	Male	Retired	Yes	No	No	No	0	1	0	0	0
14	44	Male	Waiter	Yes	Yes	Yes	Yes	0	1	1	1	1
15	18	Male	Student	Yes	Yes	Yes	No	0	1	1	1	0
16	58	Male	Lawyer	Yes	No	Yes	No	0	1	0	1	0
17	33	Female	Medical Docto	Yes	Yes	No	Yes	1	1	1	0	1
18	65	Female	Retired	No	Yes	No	Yes	1	0	1	0	1
19	24	Female	Student	No	Yes	Yes	Yes	1	0	1	1	1
20	61	Female	Retired	Yes	Yes	No	Yes	1	1	1	0	1
21	23	Female	Student	Yes	Yes	Yes	Yes	1	1	1	1	1
22	59	Female	Medical Docto	No	No	Yes	No	1	0	0	1	0
23	56	Female	Teacher	Yes	Yes	Yes	No	1	1	1	1	0
24	26	Female	Student	No	Yes	No	Yes	1	0	1	0	1
25	34	Female	Teacher	No	Yes	No	Yes	1	0	1	0	1
26	48	Female	Teacher	Yes	No	No	Yes	1	1	0	0	1
27	49	Female	Medical Docto	No	Yes	No	Yes	1	0	1	0	1
28	18	Female	Student	Yes	Yes	Yes	Yes	1	1	1	1	1
29	44	Female	Engineer	Yes	Yes	No	Yes	1	1	1	0	1
30	32	Female	Medical Docto	Yes	Yes	Yes	No	1	1	1	1	0
31	20	Female	Student	Yes	Yes	Yes	Yes	1	1	1	1	1
32	50	Female	Nurse	No	Yes	Yes	No	1	0	1	1	0
33	38	Female	Lawyer	Yes	Yes	No	Yes	1	1	1	0	1
34	58	Female	Retired	No	Yes	Yes	No	1	0	1	1	0

K-Nearest Neighbor (kNN)

- So we can use the following variables as inputs in the distance metric:

  - Age (Numerical)

  - Sex (Binary)

  - Likes Banana (Binary)

  - Likes Chocolate (Binary)

  - Lactose Intolerant (Binary)

- And we can use the distances for selecting neighbors.

K-Nearest Neighbor (kNN)

- Is our sample representative?

  - 4 binary variables and a numerical variable.

  - $2^4 = 16$ variations due to binary variables.

  - 34 samples.   Not very well for all these variables.

- Maybe we should also try with less variables.

  - More significant variables could exist.

K-Nearest Neighbor (kNN)

- Just to understand our training data,

  - We will calculate simple correlation coefficients for these variables, including the binary ones.

  - Normally we should not do this for binary data.

- Result?

  - Not a bad training data set.

  - Not necessarily very good either.

	Age	Binary Sex	Binary Banana	Binary Chocolate	Binary Lactose	Binary Milkshake
Age	1,000	0,094	-0,088	-0,214	-0,071	-0,286
Binary Sex		1,000	-0,274	0,311	0,126	0,167
Binary Banana			1,000	-0,087	-0,019	0,147
Binary Chocolate				1,000	0,074	**0,461**
Binary Lactose					1,000	**-0,459**
Binary Milkshake						1,000

K-Nearest Neighbor (kNN)

- Single Test Subject.

  - Age: 42, Male, Likes bananas, Likes chocolate, Not lactose intolerant.

- Steps in prediction:

  - Step 1. Calculate Distances.

  - Step 2. Search for k nearest neighbors (k=2,3,4,5)

  - Step 3. Predict.

K-Nearest Neighbor (kNN)

- We also have a problem with scaling.

  - After **linear scaling** the age ranges.

	Age	Binary Sex	Binary Banana	Binary Chocolate	Binary Lactose	
Test Subject	42	0	1	1	0	

No	Age Distance	Sex Distance	Banana Distance	Chocolage Distance	Lactose Distance	Total Distance
1	0,149	0,000	0,000	0,000	0,000	0,149
2	0,128	0,000	0,000	1,000	0,000	1,128
3	0,149	0,000	1,000	1,000	0,000	2,149
4	0,064	0,000	1,000	0,000	1,000	2,064
5	0,106	0,000	0,000	0,000	1,000	1,106
6	0,043	0,000	0,000	0,000	0,000	0,043
7	0,383	0,000	0,000	1,000	0,000	1,383
8	0,340	0,000	0,000	0,000	0,000	0,340
9	0,511	0,000	1,000	0,000	0,000	1,511
10	0,489	0,000	0,000	0,000	0,000	0,489
11	0,170	0,000	0,000	0,000	0,000	0,170
12	0,021	0,000	0,000	1,000	1,000	2,021
13	0,362	0,000	0,000	1,000	0,000	1,362
14	0,043	0,000	0,000	0,000	1,000	1,043
15	0,511	0,000	0,000	0,000	1,000	1,511
16	0,340	0,000	0,000	1,000	1,000	2,340
17	0,191	1,000	0,000	0,000	0,000	1,191
18	0,489	1,000	1,000	0,000	0,000	2,489
19	0,383	1,000	1,000	0,000	1,000	3,383
20	0,404	1,000	0,000	0,000	0,000	1,404
21	0,404	1,000	0,000	0,000	1,000	2,404
22	0,362	1,000	1,000	1,000	1,000	4,362
23	0,298	1,000	0,000	0,000	1,000	2,298
24	0,340	1,000	1,000	0,000	0,000	2,340
25	0,170	1,000	0,000	1,000	0,000	2,170
26	0,128	1,000	0,000	1,000	0,000	2,128
27	0,149	1,000	1,000	0,000	0,000	2,149
28	0,511	1,000	0,000	0,000	1,000	2,511
29	0,043	1,000	0,000	0,000	0,000	1,043
30	0,213	1,000	0,000	0,000	1,000	2,213
31	0,468	1,000	0,000	0,000	1,000	2,468
32	0,170	1,000	1,000	0,000	1,000	3,170
33	0,085	1,000	0,000	0,000	0,000	1,085
34	0,340	1,000	1,000	0,000	1,000	3,340

K-Nearest Neighbor (kNN)

No	Age Distance	Sex Distance	Banana Distance	Chocolage Distance	Lactose Distance	Total Distance	Binary Milkshake
6	0,043	0,000	0,000	0,000	0,000	0,043	1
1	0,149	0,000	0,000	0,000	0,000	0,149	1
11	0,170	0,000	0,000	0,000	0,000	0,170	0
8	0,340	0,000	0,000	0,000	0,000	0,340	1
10	0,489	0,000	0,000	0,000	0,000	0,489	1
14	0,043	0,000	0,000	0,000	1,000	1,043	0
29	0,043	1,000	0,000	0,000	0,000	1,043	0
33	0,085	1,000	0,000	0,000	0,000	1,085	0
5	0,106	0,000	0,000	0,000	1,000	1,106	1
2	0,128	0,000	0,000	1,000	0,000	1,128	1
17	0,191	1,000	0,000	0,000	0,000	1,191	1
13	0,362	0,000	0,000	1,000	0,000	1,362	0
7	0,383	0,000	0,000	1,000	0,000	1,383	1
20	0,404	1,000	0,000	0,000	0,000	1,404	0
9	0,511	0,000	1,000	0,000	0,000	1,511	0
15	0,511	0,000	0,000	0,000	1,000	1,511	0
12	0,021	0,000	0,000	1,000	1,000	2,021	1
4	0,064	0,000	1,000	0,000	1,000	2,064	1
26	0,128	1,000	0,000	1,000	0,000	2,128	1
3	0,149	0,000	1,000	1,000	0,000	2,149	1
27	0,149	1,000	1,000	0,000	0,000	2,149	0
25	0,170	1,000	1,000	0,000	0,000	2,170	1
30	0,213	1,000	0,000	0,000	1,000	2,213	1
23	0,298	1,000	0,000	0,000	1,000	2,298	1
16	0,340	0,000	0,000	1,000	1,000	2,340	0
24	0,340	1,000	1,000	0,000	0,000	2,340	1
21	0,404	1,000	0,000	0,000	1,000	2,404	0
31	0,468	1,000	0,000	0,000	1,000	2,468	1
18	0,489	1,000	1,000	0,000	0,000	2,489	1
28	0,511	1,000	0,000	0,000	1,000	2,511	1
32	0,170	1,000	1,000	0,000	1,000	3,170	1
34	0,340	1,000	1,000	0,000	1,000	3,340	1
19	0,383	1,000	1,000	0,000	1,000	3,383	1
22	0,362	1,000	1,000	1,000	1,000	4,362	0

K-Nearest Neighbor (kNN)

k	Binary Majority	Result
2	1 (by %100)	Likes
3	1 (by %67)	Likes
4	1 (by %75)	Likes
5	1 (by %80)	Likes
6	1 (by %67)	Likes

· Really nice, but is it that conclusive for all test subjects?
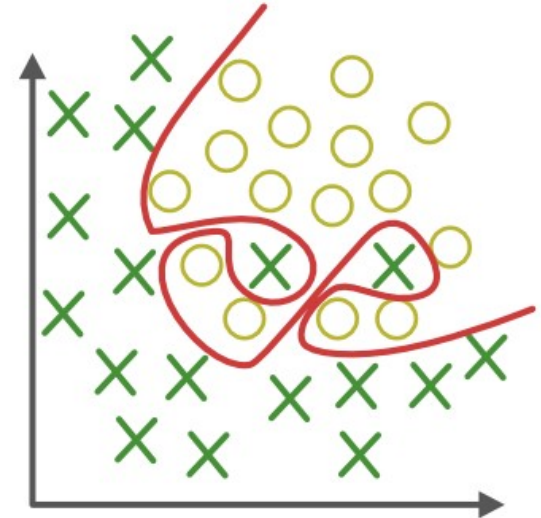
· How should we evaluate our results?

K-Nearest Neighbor (kNN)
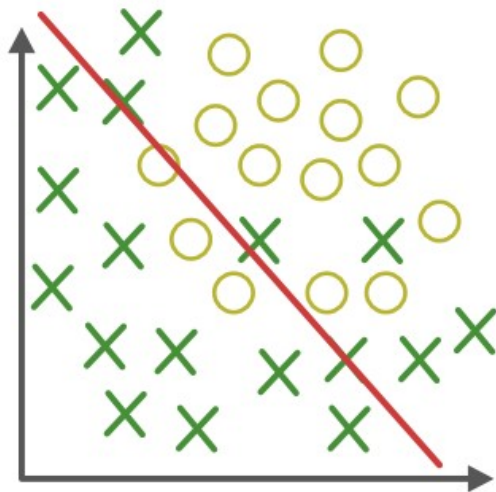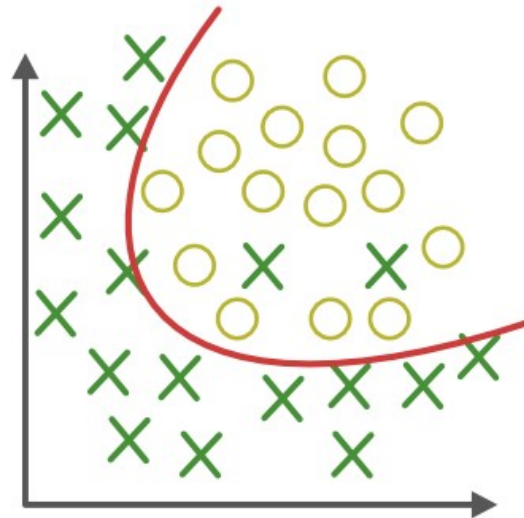
K-Nearest Neighbor (kNN)

- **Overfitting** occurs when a statistical model fits exactly against its training data.

  - When the model memorizes the noise and fits too closely to the training set, the model becomes "overfitted," and it is unable to generalize well to new data.

  - If a model cannot generalize well to new data, then it will not be able to perform the classification or prediction tasks that it was intended for.

- Low error rates and a high variance are good indicators of overfitting.

- If the training data has a low error rate and the test data has a high error rate, it signals overfitting.

K-Nearest Neighbor (kNN)
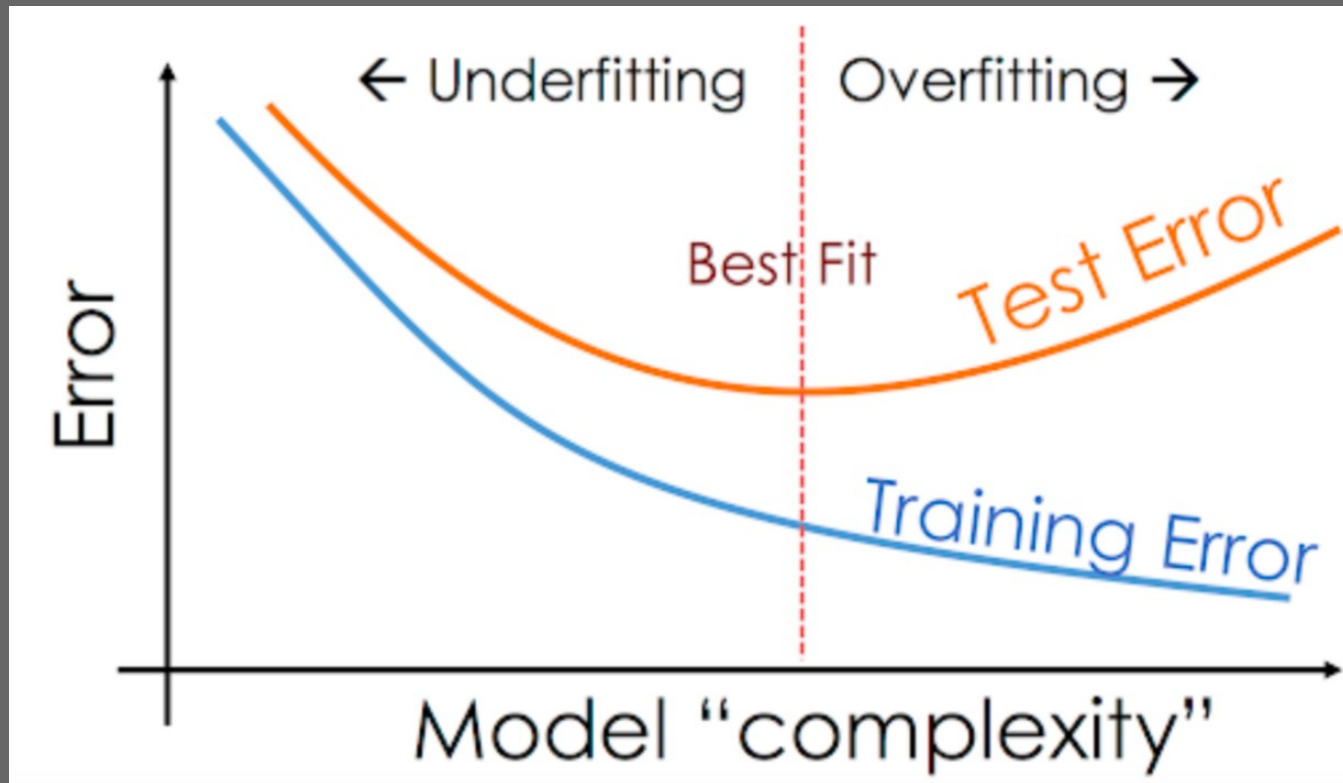
K-Nearest Neighbor (kNN)

- We should be able to **detect overfitting** so that we can avoid it before our models are used to make actual predictions (ie. classifications).

- Recall s-fold-cross-validation.

  - Calculate **performance metrics** for each run.

  - Observe the **variance** of these metrics.
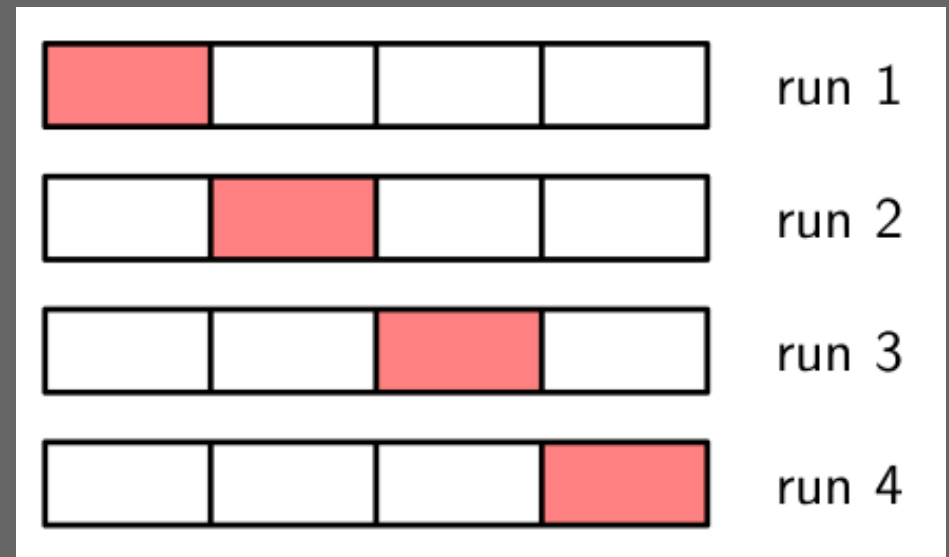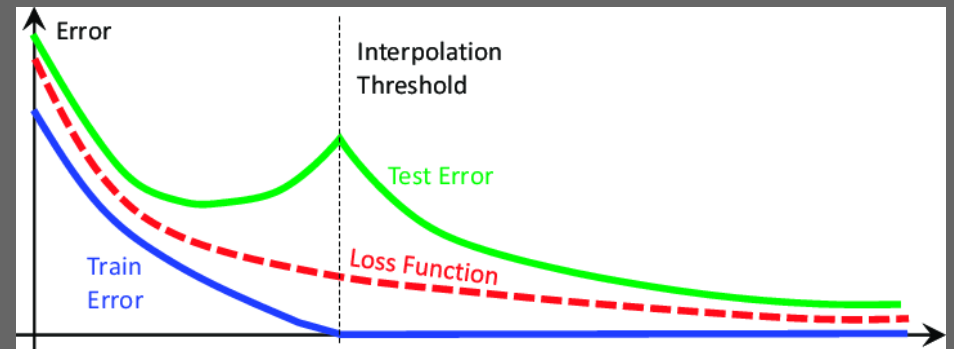
K-Nearest Neighbor (kNN)

- Can we **avoid overfitting**? There is no exact solution, but there are some techniques.

  - **Early stopping** pauses the training before the model starts learning the noise within the model. This approach risks halting the training process too soon, leading to the opposite problem of underfitting.

  - **Data augmentation** adds noisy data is to make a more stable model. In theory the added noise will make it **harder to overfit to the existing noise**. However, there is no reliable way to manage this approach.

K-Nearest Neighbor (kNN)

- Can we **avoid overfitting**? There is no exact solution, but there are some techniques.

  - **Feature selection** is the process of identifying the most important ones within the training data and then eliminating the irrelevant or redundant ones. This is commonly mistaken for dimensionality reduction, but it is different.

  - **Regularization** applies a penalty to the input parameters with the larger coefficients, which subsequently limits the amount of variance in the model. These methods aim to **identify and reduce the noise within the data**. They are also classified under

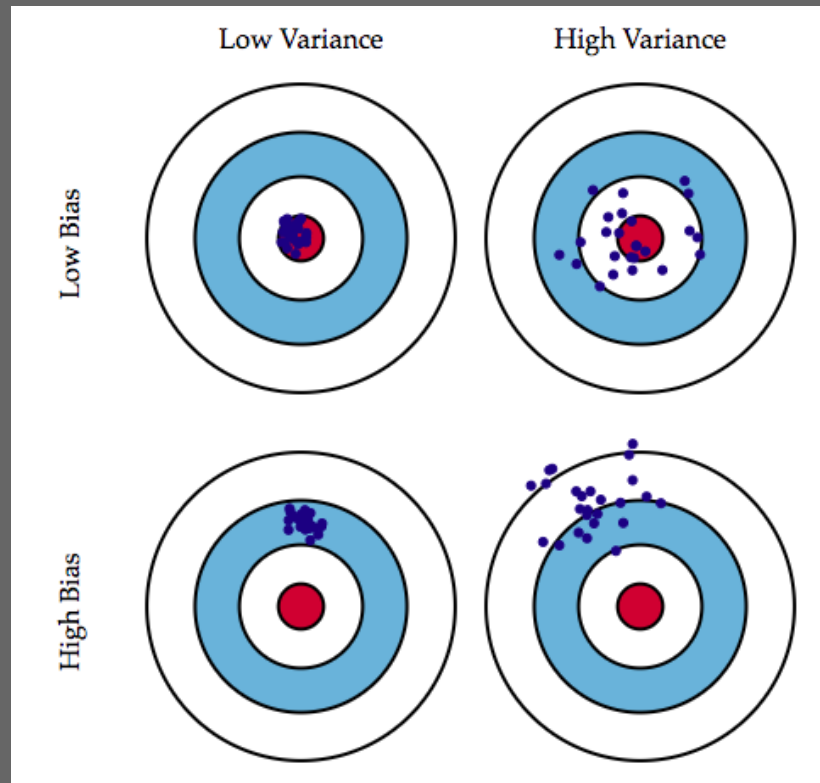# K-Nearest Neighbor (kNN)

- Can we **avoid overfitting**? .

  - If you can use hyper-parameter optimizations very successfully, you might end up with a **double decent curve** such as the one presented.

  - Most deep learning models **promise** this. What they deliver of course varies.
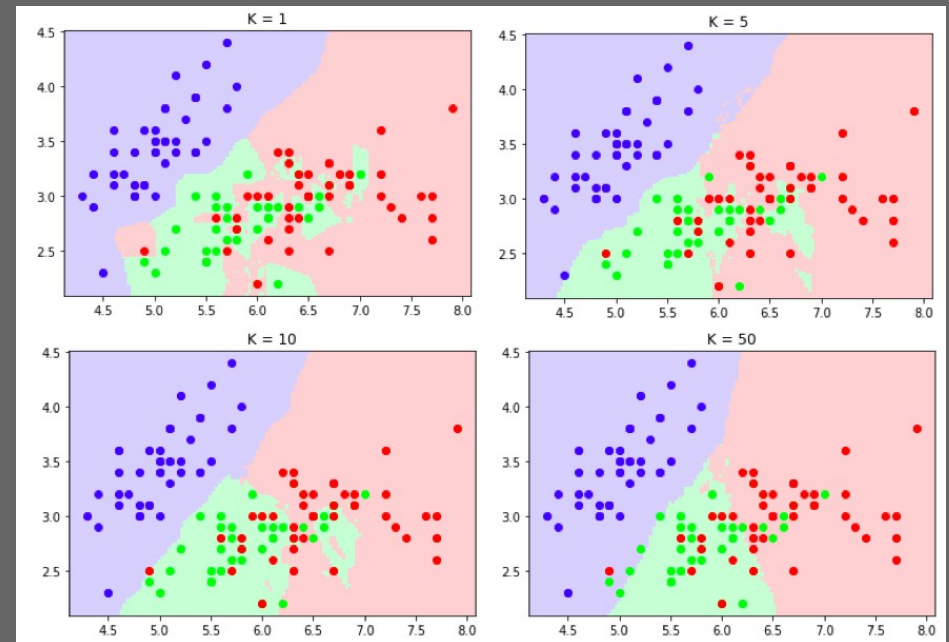
K-Nearest Neighbor (kNN)

- Bias – Variance Trade-off
  - Bias is a type error from wrong assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between selected inputs and target outputs.
  - Variance is an error from sensitivity to small fluctuations in the training set. High variance can cause an algorithm to model the random noise in the training data instead of actual outputs.

K-Nearest Neighbor (kNN)

K-Nearest Neighbor (kNN)

- Regarding kNN, there exists a "decision boundary" which defines the result based on k, irrespective of the input.

  - Why irrespective?

  - Is this a measure for "a good k?"

- Sample code demonstrating decision boundary with a very nice explanation

  - https://medium.com/30-days-of-machine-learning/day-3-k-nearest-neighbors-and-bias-variance-tradeoff-75f84d515bdb

Questions?

CONTACT:
bora.gungoren@atilim.edu.tr