RANJAN2612 (HTTPS://ID.ANALYTICSVIDHYA.COM/ACCOUNTS/PROFILE/)

Analytics Vidhya
Learn everything about analytics
(https://www.analyticsvidhya.com/blog/)

BIG DATA (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/BIG-DATA/)
BUSINESS ANALYTICS (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/BUSINESS-ANALYTICS/)
MACHINE LEARNING (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/MACHINE-LEARNING/)
PYTHON (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/PYTHON-2/)

Introduction to k-Nearest Neighbors: Simplified (with implementation in Python)

TAVISH SRIVASTAVA (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/AUTHOR/TAVISH1/), MARCH 26, 2018

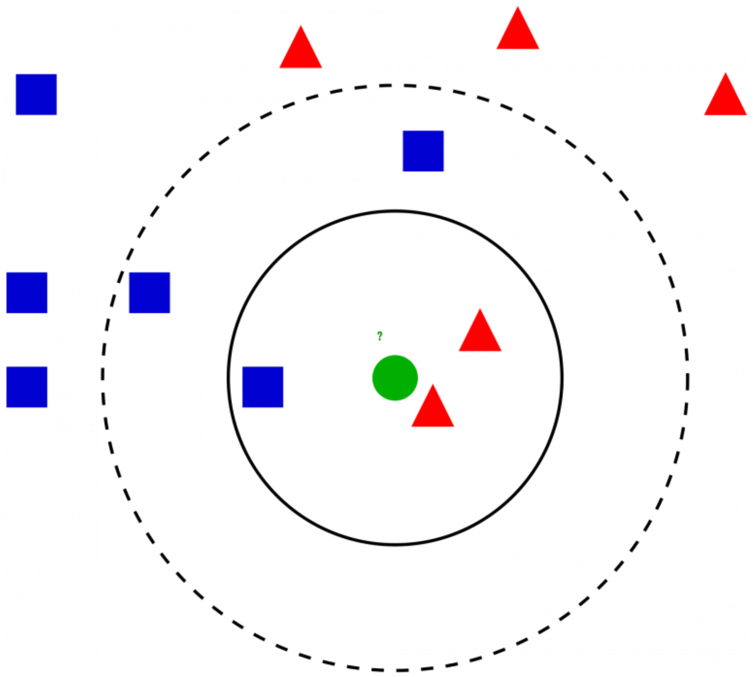
Note: This article was originally published on Oct 10, 2014 and updated on Mar 27th, 2018

Introduction

In the four years of my data science career (https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2/?utm_source=blog&utm_medium=introknearestneighborarticle), I have built more than 80% classification models and just 15-20% regression models. These ratios can be more or less generalized throughout the industry. The reason behind this bias towards classification models (https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2/?utm_source=blog&utm_medium=introknearestneighborarticle) is that most analytical problems involve making a decision.

For instance, will a customer attrite or not, should we target customer X for digital campaigns, whether customer has a high potential or not etc. These analysis are more insightful and directly linked to an implementation roadmap.

Download Resource



(https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2018/03/knn3.png)

In this article, we will talk about another widely used classification techniqu (https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2/?utm_source=blog&utm_medium=introknearestneighborarticle)e called K-nearest neighbors (KNN) . Our focus will be primarily on how does the algorithm work and how does the input parameter effect the output/prediction.

Table of Contents

When do we use KNN algorithm?

KNN can be used for both classification and regression predictive problems. However, it is more widely used in classification problems in the industry. To evaluate any technique we generally look at 3 important aspects:

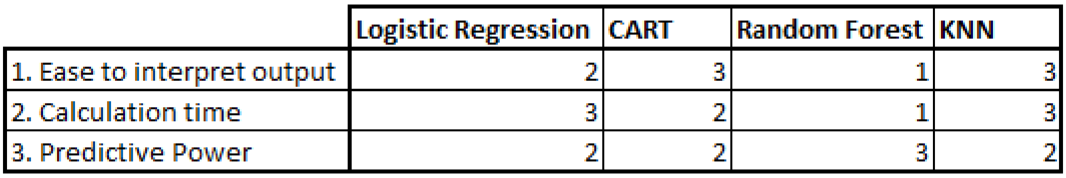
1. Ease to interpret output

Download Resource

2. Calculation time

3. Predictive Power

Let us take a few examples to place KNN in the scale :

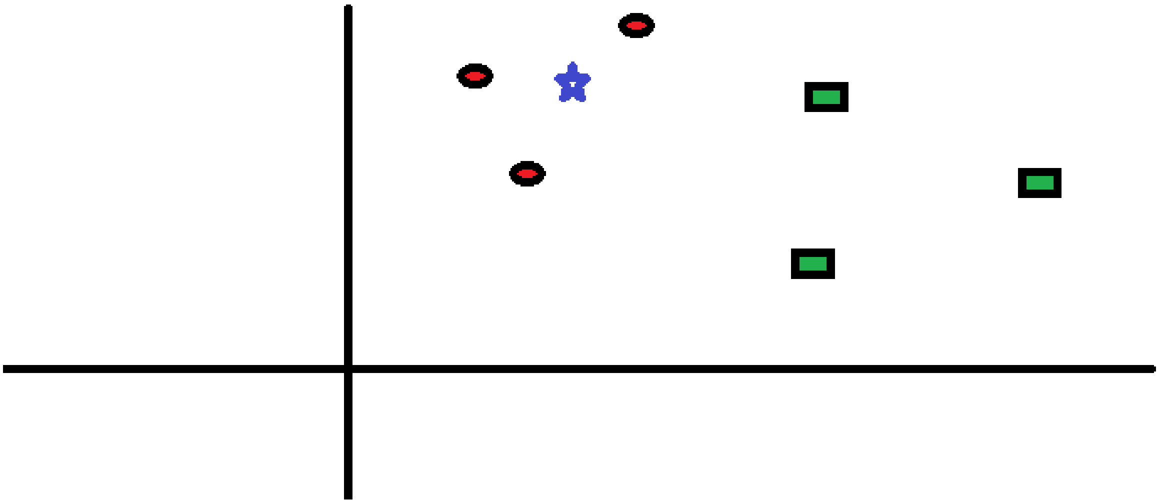


	Logistic Regression	CART	Random Forest	KNN
1. Ease to interpret output	2	3	1	3
2. Calculation time	3	2	1	3
3. Predictive Power	2	2	3	2

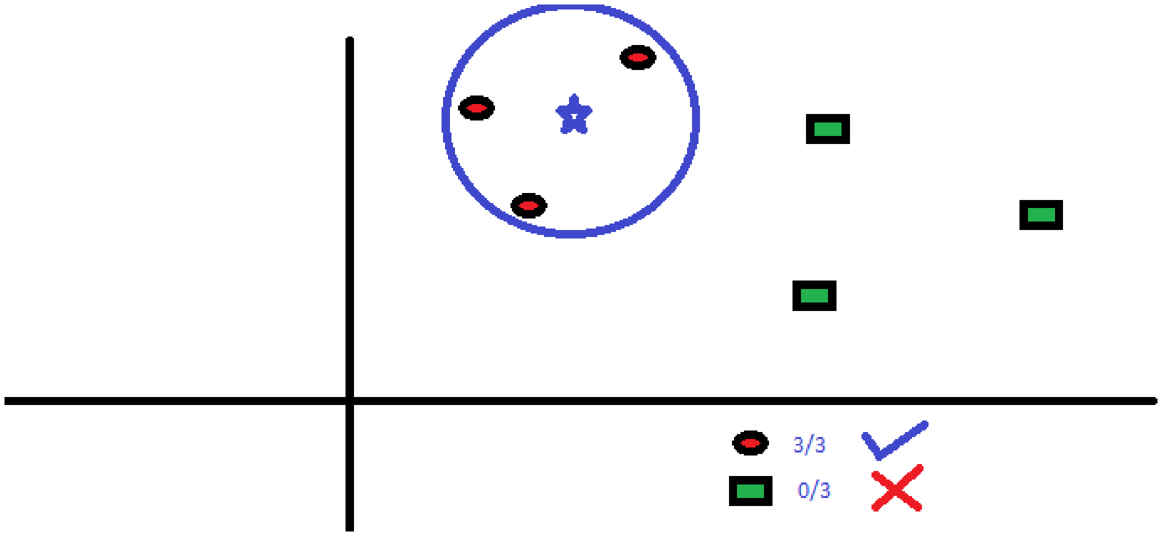
(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/Model-comparison.png)KNN algorithm fairs across all parameters of considerations. It is commonly used for its easy of interpretation and low calculation time.

How does the KNN algorithm work?

Let's take a simple case to understand this algorithm. Following is a spread of red circles (RC) and green squares (GS) :



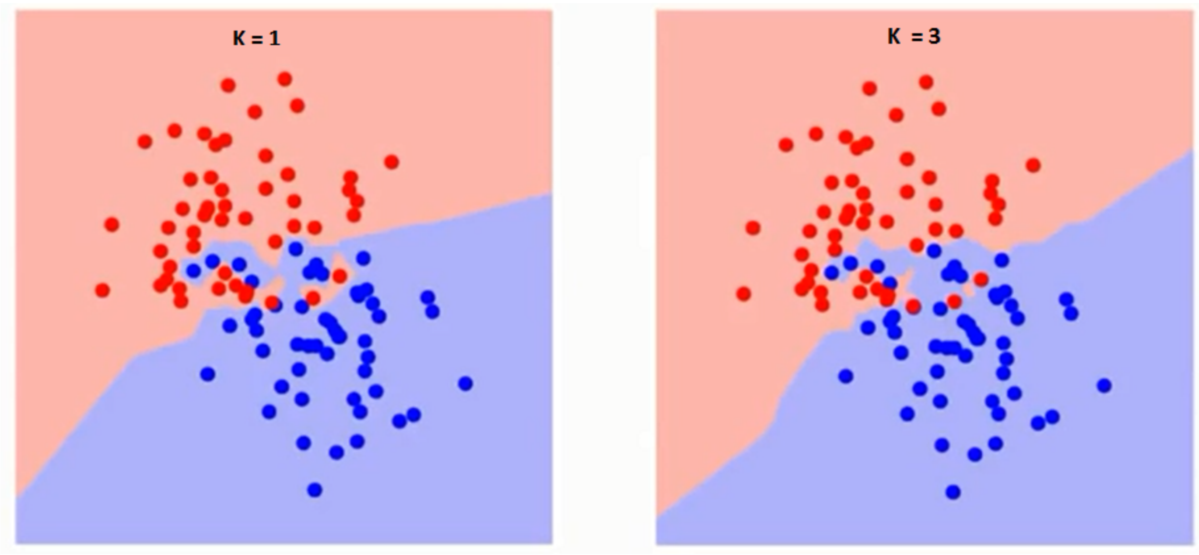
(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/scenario1.png)You intend to find out the class of the blue star (BS) . BS can either be RC or GS and nothing else. The "K" is KNN algorithm is the nearest neighbors we wish to take vote from. Let's say K = 3. Hence, we will now make a circle with BS as center just as big as to enclose only three datapoints on the plane. Refer to following diagram for more details:



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/scenario2.png) The three closest points to BS is all RC. Hence, with good confidence level we can say that the BS should belong to the class RC. Here, the choice became very obvious as all three votes from the closest neighbor went to RC. The choice of the parameter K is very crucial in this algorithm. Next we will understand what are the factors to be considered to conclude the best K.

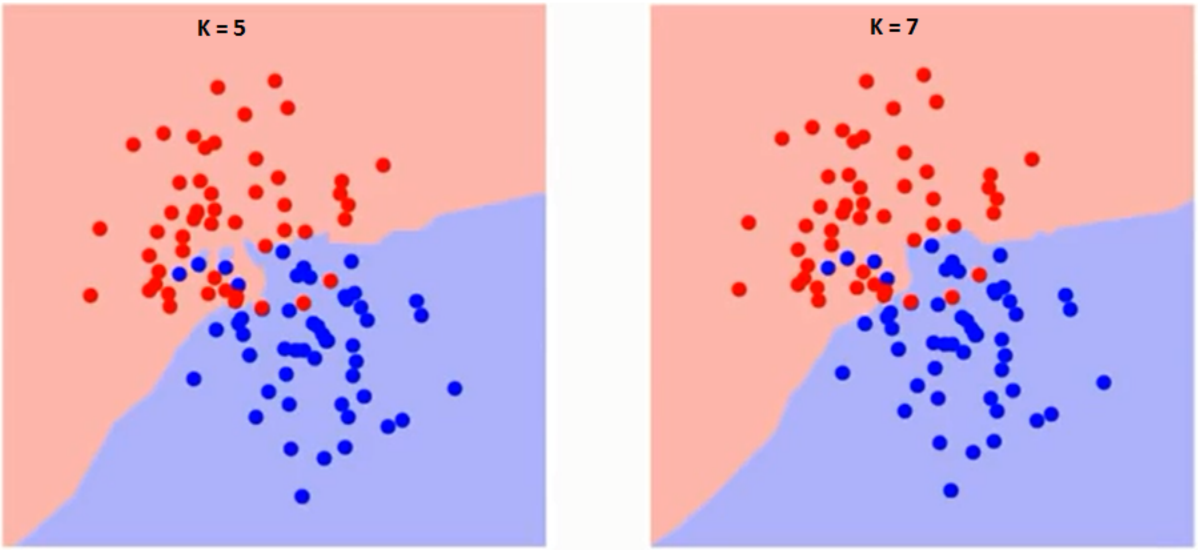
How do we choose the factor K?

First let us try to understand what exactly does K influence in the algorithm. If we see the last example, given that all the 6 training observation remain constant, with a given K value we can make boundaries of each class. These boundaries will segregate RC from GS. The same way, let's try to see the effect of value "K" on the class boundaries. Following are the different boundaries separating the two classes with different values of K.



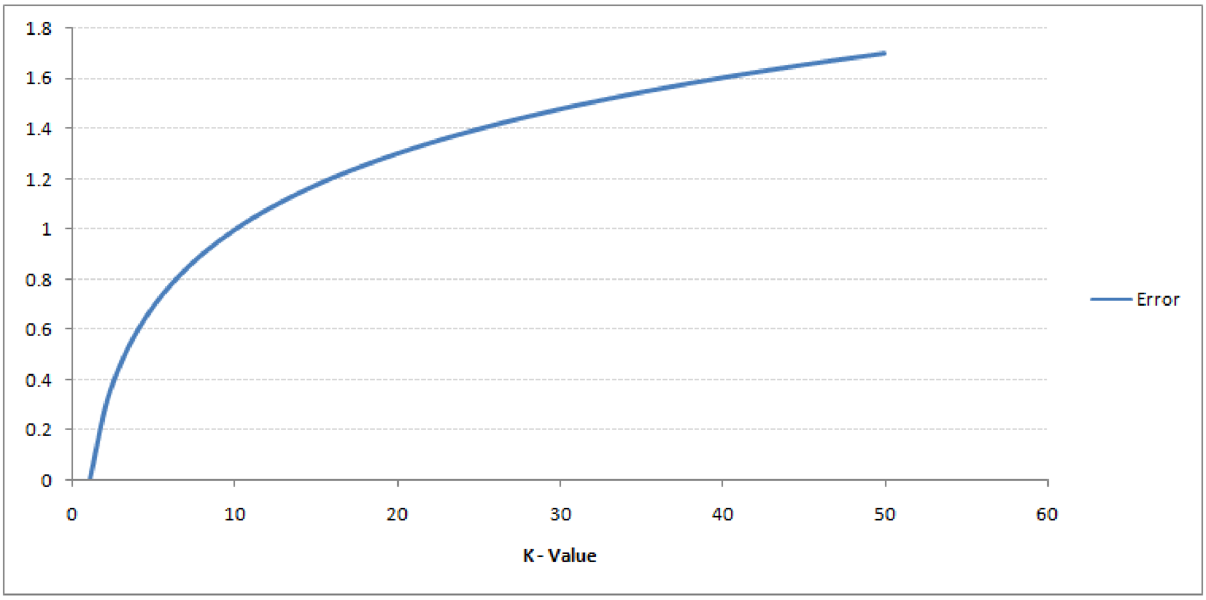
(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/K-judgement.png)

Download Resource



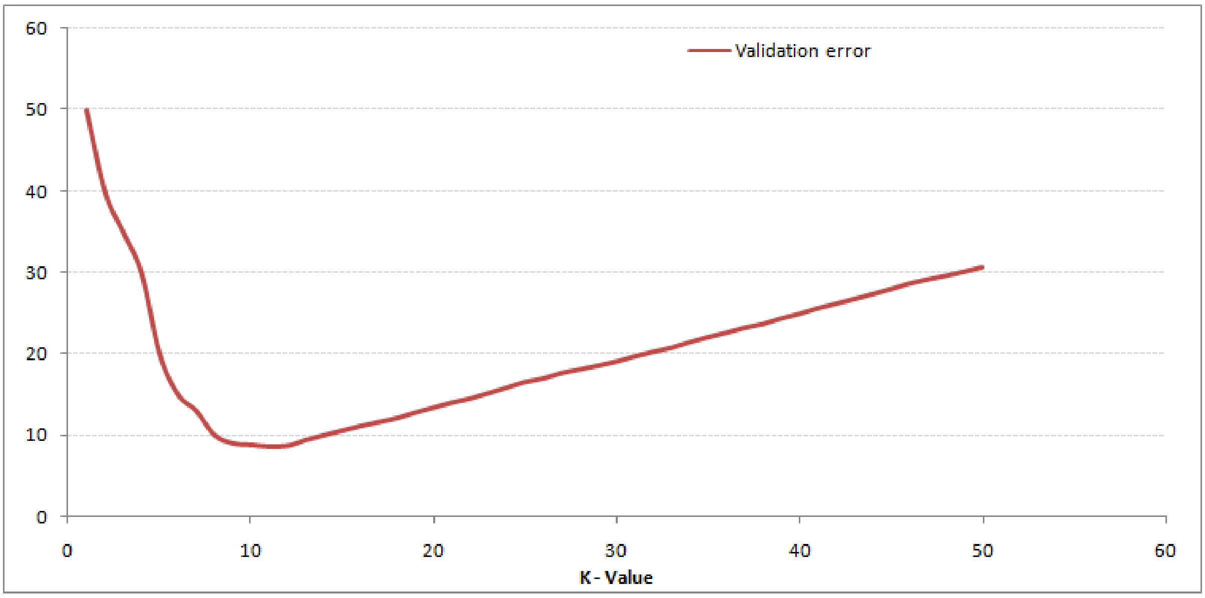
(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/K-judgement2.png)

If you watch carefully, you can see that the boundary becomes smoother with increasing value of K. With K increasing to infinity it finally becomes all blue or all red depending on the total majority. The training error rate and the validation error rate are two parameters we need to access on different K-value. Following is the curve for the training error rate with varying value of K :



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/training-error.png)As you can see, the error rate at K=1 is always zero for the training sample. This is because the closest point to any training data point is itself.Hence the prediction is always accurate with K=1. If validation error curve would have been similar, our choice of K would have been 1. Following is the validation error curve with varying value of K:

Download Resource



(https://www.analyticsvidhya.com/blog/wp-content/uploads/2014/10/training-error_11.png)This makes the story more clear. At K=1, we were overfitting the boundaries. Hence, error rate initially decreases and reaches a minima. After the minima point, it then increase with increasing K. To get the optimal value of K, you can segregate the training and validation from the initial dataset. Now plot the validation error curve to get the optimal value of K. This value of K should be used for all predictions.

Breaking it Down – Pseudo Code of KNN

We can implement a KNN model by following the below steps:

1. Load the data
2. Initialise the value of k
3. For getting the predicted class, iterate from 1 to total number of training data points
    1. Calculate the distance between test data and each row of training data. Here we will use Euclidean distance as our distance metric since it's the most popular method. The other metrics that can be used are Chebyshev, cosine, etc.
    2. Sort the calculated distances in ascending order based on distance values
    3. Get top k rows from the sorted array
    4. Get the most frequent class of these rows
    5. Return the predicted class

Implementation in Python from scratch

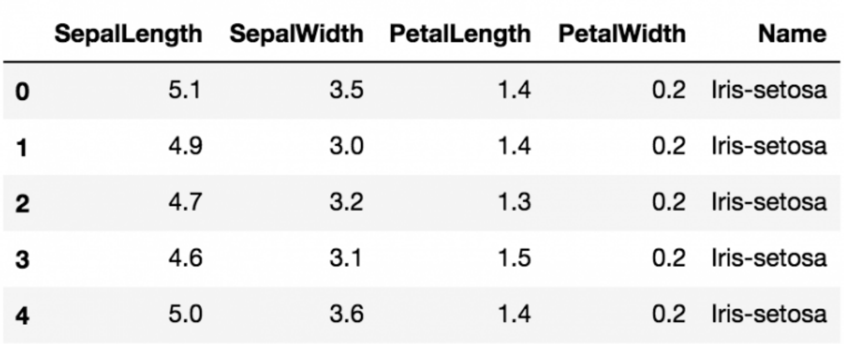
We will be using the popular Iris dataset for building our KNN model. You can download it from here (https://gist.githubusercontent.com/gurchetan1000/ec90a0a8004927e57c24b20a6f8c8d35/raw/fcd83b35021

Download Resource

```
# Importing libraries
import pandas as pd
import numpy as np
import math
import operator
```

```
#### Start of STEP 1
# Importing data
data = pd.read_csv("iris.csv")
#### End of STEP 1

data.head()
```



	SepalLength	SepalWidth	PetalLength	PetalWidth	Name
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

Download Resource

```
# Defining a function which calculates euclidean distance between two data points
def euclideanDistance(data1, data2, length):
    distance = 0
    for x in range(length):
        distance += np.square(data1[x] - data2[x])
    return np.sqrt(distance)

# Defining our KNN model
def knn(trainingSet, testInstance, k):

    distances = {}
    sort = {}

    length = testInstance.shape[1]

    #### Start of STEP 3
    # Calculating euclidean distance between each row of training data and test data
    for x in range(len(trainingSet)):

        #### Start of STEP 3.1
        dist = euclideanDistance(testInstance, trainingSet.iloc[x], length)

        distances[x] = dist[0]
        #### End of STEP 3.1

    #### Start of STEP 3.2
    # Sorting them on the basis of distance
    sorted_d = sorted(distances.items(), key=operator.itemgetter(1))
    #### End of STEP 3.2

    neighbors = []

    #### Start of STEP 3.3
    # Extracting top k neighbors
    for x in range(k):
        neighbors.append(sorted_d[x][0])
```

Download Resource

```
    #### End of STEP 3.3
    classVotes = {}

    #### Start of STEP 3.4
    # Calculating the most freq class in the neighbors
    for x in range(len(neighbors)):
        response = trainingSet.iloc[neighbors[x]][-1]

        if response in classVotes:
            classVotes[response] += 1
        else:
            classVotes[response] = 1
    #### End of STEP 3.4

    #### Start of STEP 3.5
    sortedVotes = sorted(classVotes.items(), key=operator.itemgetter(1), reverse=True)
    return(sortedVotes[0][0], neighbors)
    #### End of STEP 3.5
```

```
# Creating a dummy testset
testSet = [[7.2, 3.6, 5.1, 2.5]]
test = pd.DataFrame(testSet)
```

```
#### Start of STEP 2
# Setting number of neighbors = 1
k = 1
#### End of STEP 2
# Running KNN model
result,neigh = knn(data, test, k)

# Predicted class
print(result)
-> Iris-virginica
```

```
# Nearest neighbor
print(neigh)
-> [141]
```

Now we will try to alter the *k* values, and see how the prediction changes.

```
# Setting number of neighbors = 3
k = 3
# Running KNN model
result,neigh = knn(data, test, k)
# Predicted class
print(result) -> Iris-virginica
```

```
# 3 nearest neighbors
print(neigh)
-> [141, 139, 120]
```

```
# Setting number of neighbors = 5
k = 5
# Running KNN model
result,neigh = knn(data, test, k)
# Predicted class
print(result) -> Iris-virginica
```

```
# 5 nearest neighbors
print(neigh)
-> [141, 139, 120, 145, 144]
```

Comparing our model with scikit-learn

```
from sklearn.neighbors import KNeighborsClassifier
neigh = KNeighborsClassifier(n_neighbors=3)
neigh.fit(data.iloc[:,0:4], data['Name'])

# Predicted class
print(neigh.predict(test))

-> ['Iris-virginica']

# 3 nearest neighbors
print(neigh.kneighbors(test)[1])
-> [[141 139 120]]
```

We can see that both the models predicted the same class ('Iris-virginica') and the same nearest neighbors ([141 139 120]). Hence we can conclude that our model runs as expected.

End Notes

KNN algorithm is one of the simplest classification algorithm. Even with such simplicity, it can give highly competitive results. KNN algorithm can also be used for regression problems. The only difference from the discussed methodology will be using averages of nearest neighbors rather than voting from nearest neighbors. KNN can be coded in a single line on R. I am yet to explore how can we use KNN algorithm on SAS.

Did you find the article useful? Have you used any other machine learning tool recently? Do you plan to use KNN in any of your business problems? If yes, share with us how you plan to go about it.

If you like what you just read & want to continue your analytics learning, subscribe to our emails (http://feedburner.google.com/fb/a/mailverify?uri=analyticsvidhya), follow us on twitter (http://twitter.com/analyticsvidhya) or like our facebook page (http://facebook.com/analyticsvidhya).

Download Resource

Share this:

(https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/?share=linkedin&nb=1)

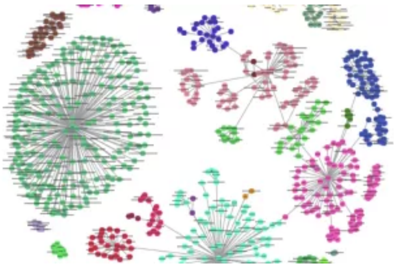
(https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/?share=facebook&nb=1)

(https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/?share=twitter&nb=1)

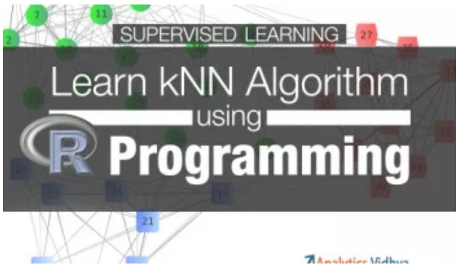
(https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/?share=pocket&nb=1)

(https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/?share=reddit&nb=1)

Related Articles



(https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/)
A Practical Introduction to K-Nearest Neighbors Algorithm for Regression (with Python code)
(https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/)
August 22, 2018
In "Machine Learning"



(https://www.analyticsvidhya.com/blog/2015/08/learning-concept-knn-algorithms-programming/)
Best way to learn kNN Algorithm using R Programming
(https://www.analyticsvidhya.com/blog/2015/08/learning-concept-knn-algorithms-programming/)
August 19, 2015
In "Business Analytics"

(https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/)
30 Questions to test a data scientist on K-Nearest Neighbors (kNN) Algorithm
(https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/)
September 4, 2017
In "Machine Learning"

TAGS : K NEAREST (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/K-NEAREST/), KNN (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/KNN/), KNN FROM SCRATCH (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/KNN-FROM-SCRATCH/), MACHINE LEARNING

Download Resource

(HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/MACHINE-LEARNING/), SIMPLIED SERIES (HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/TAG/SIMPLIED-SERIES/)

NEXT ARTICLE

AVBytes: AI & ML Developments this week – IBM's Library 46 Times Faster than TensorFlow, Baidu's Massive Self-Driving Dataset, the Technology behind AWS SageMaker, etc.

(https://www.analyticsvidhya.com/blog/2018/03/avbytes-ai-ml-developments-this-week-260318/)

•••

PREVIOUS ARTICLE

DeepMind is Using 'Neuron Deletion' to Understand Deep Neural Networks

(https://www.analyticsvidhya.com/blog/2018/03/deepmind-using-neuron-deleting-understand-deep-neural-networks/)

(https://www.analyticsvidhya.com/blog/author/tavish1/)

Tavish Srivastava (Https://Www.Analyticsvidhya.Com/Blog/Author/Tavish1/)

Tavish is an IIT post graduate, a results-driven analytics professional and a motivated leader with 7+ years of experience in data science industry. He has led various high performing data scientists teams in financial domain. His work range from creating high level business strategy for customer engagement and acquisition to developing Next-Gen cognitive Deep/Machine Learning capabilities aligned to these high level strategies for multiple domains including Retail Banking, Credit Cards and Insurance. Tavish is fascinated by the idea of artificial intelligence inspired by human intelligence and enjoys every discussion, theory or even movie related to this idea.

This article is quite old and you might not get a prompt response from the author. We request you to post this comment on Analytics Vidhya's **Discussion portal** (https://discuss.analyticsvidhya.com/) to get

Download Resource

your queries resolved

35 COMMENTS

HARSHAL

Reply

October 10, 2014 at 3:29 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27697)

Useful article.
Can you share similar article for randomforest ?
What are limitations with data size for accuracy?

TAVISH SRIVASTAVA

Reply

October 10, 2014 at 4:49 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27709)

Harshal,
We have already published many articles on random forest. Here is the link of the article on random forest on similar lines http://www.analyticsvidhya.com/blog/2014/06/introduction-random-forest-simplified/ (http://www.analyticsvidhya.com/blog/2014/06/introduction-random-forest-simplified/).
You can also subscribe to analyticsvidhya to get access to weekly updates on such articles.

SAURABH

Reply

October 10, 2014 at 5:00 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27710)

Good one please share the value of
Red circle and green square

TAVISH SRIVASTAVA

Reply

October 10, 2014 at 9:51 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27813)

Saurabh,

The first graph is for illustrating purposes. You can create a random dataset to check the algorithm.

DEBASHIS ROUT

Reply

October 10, 2014 at 7:02 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27738)

I am currently doing part time MS in BI & Data Mining. I found this article is really helpful to understand in more detail and expecting to utilize in my upcoming project work. I need to know do you have any article on importance of Data quality in BI , Classification & Decision Tree.

TAVISH

Reply

October 10, 2014 at 9:48 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27808)

Debashish,
We have published many articles on CART models before. Here is a link which will give you a kick start http://www.analyticsvidhya.com/blog/2014/06/comparing-random-forest-simple-cart-model/ (http://www.analyticsvidhya.com/blog/2014/06/comparing-random-forest-simple-cart-model/).

SARASWATHI

Reply

October 10, 2014 at 1:44 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27943)

Hello
the article is very clear and precise. I would like some clarification on the single line
“To get the optimal value of K, you can segregate the training and validation from the initial dataset.”. Do you mean that we segregate those points on the border of the boundaries for validation and keep the remaining for training. This is not very clear to me. Can you please elaborate ?

Thanks.

TAVISH SRIVASTAVA

Reply

October 10, 2014 at 3:38 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27994)

Saraswathi,

Here is what I meant : Take the entire population, and randomly split it into two. Now on the training sample, score each validation observation with different k-values. The error curve will give you the best value of k.

Hope it becomes clear now.

SARASWATHI

Download Resource

Reply

October 10, 2014 at 4:41 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-28016)

I want to make sure I understand this correctly. Please confirm or correct.

You say, take the entire population and split into two – are these two divisions, one for training and one for validation ? (I am assuming so).

So, now I use different values of K to cluster the training samples.

I try to see where the validation samples fall in these clusters.

I draw the error curve and choose the K with the smallest error ?

TAVISH Reply

October 13, 2014 at 11:02 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-29158)

Saraswathi,
Let me make it even simpler. Say, you have 100 datapoints. Split this population into two samples of 70 and 30 observations. Use these 70 observation to predict for the other 30. Once you have the prediction for a particular value of k, check the misclassification with actual value. Repeat this exercise for different value of k. Hopefully, you will get a curve similar to that shown in the article. Now choose the k for which the misclassification is least.

Hope this makes it clear.

Tavish

HARVEY S Reply

October 10, 2014 at 1:50 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27944)

Nice tease: “KNN can be coded in a single line on R. ”

Can you give an example?

TAVISH SRIVASTAVA Reply

October 10, 2014 at 3:38 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-27995)

We will cover this piece in our coming articles. Stay tuned.

FELIX Reply

October 13, 2014 at 9:02 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-29121)

Hi, great post, thanks. I would like to add, that "low calculation time" is not true for the prediction phase with big, high dimensional datasets. But it's still a good choice in many applications.

TAVISH Reply

October 13, 2014 at 11:03 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-29162)

Felix,

You are probably right for cases where the distance between observations comparable in the large dataset. But in general population have natural clusters which makes the calculation faster. Let me know in case you disagree.

Tavish

KABIR SINGH Reply

October 15, 2014 at 8:05 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-30127)

I am trying to figure out churn analysis, any suggestions where I am start looking?

BTW following this website for 6-8 months now, you guys are doing an amazing job

NAJMA NAAZ Reply

June 10, 2015 at 5:08 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-88260)

That was very helpful. Thank you! Can you please share a concise article on neural nets and deep learning as well?

TIAGO Reply

June 21, 2016 at 6:32 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-112504)

Thank you.

CHARLES

Reply

February 12, 2017 at 6:45 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-122572)

Very useful. it was very explanatory. Thanks for that. Can you please post about adabooster algorithm?

AISHWARYA SINGH

Reply

October 8, 2018 at 7:40 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-155233)

You'll find it here : https://www.analyticsvidhya.com/blog/2015/11/quick-introduction-boosting-algorithms-machine-learning/ (https://www.analyticsvidhya.com/blog/2015/11/quick-introduction-boosting-algorithms-machine-learning/)

THUE XE DU LICH GIA RE

Reply

June 26, 2017 at 4:24 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-131138)

Quality articles or reviews is the secret to
invite the visitors to visit the website, that's what this web site is providing.

EMANUEL FAKHAR

Reply

July 23, 2017 at 8:17 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-132717)

The world of DS would be so boring and exaggerated without you guys. Anything I study, I get a better perspective from this site. And you are so generous and grounded compared to idiots here in UK. God bless.

STONEHEAD PARK

Reply

September 24, 2017 at 10:45 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-137835)

Excellent post, I appreciate your effort. 🙂

JUST81100

Reply

Download Resource

September 28, 2017 at 12:23 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-138166)

KNN is fast to train but the prediction speed grows exponentially with the data set size and his complexity rather than Random forest…

SOUMYA SHREYA **Reply**

March 27, 2018 at 7:00 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152214)

It is a really nice and well explained article, I am a beginner in the field of data science and machine learning and I find these articles really helpful to learn and understand the algorithms. Thanks for publishing them. Can you suggest me some datasets where I can experiment and apply KNN.

AISHWARYA SINGH **Reply**

March 29, 2018 at 1:32 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152256)

Hi Soumya,

You can use the Cancer dataset to practice kNN.
Refer this link (https://discuss.analyticsvidhya.com/t/practice-dataset-for-knn-algorithm/3104) for the same.

KRISHNA **Reply**

March 27, 2018 at 7:04 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152215)

How do we handle categorical features with kNN? Do we need to create dummies for them? Do you suggest any other distance method other than euclidean distance if we have more number of features? I feel we have to treat outliers as they may have impact on the distances, similarly missing values.. Please share your opinion.

AISHWARYA SINGH **Reply**

March 28, 2018 at 3:21 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152229)

Hi,

Download Resource

Yes you can create dummies for categorical variables in kNN.
Apart from Euclidean distance, there are other methods that can be used to find the distance such as Manhattan or Minkowski.
For outliers adn missing value treatment, you can refer this article (https://www.analyticsvidhya.com/blog/2016/01/guide-data-exploration/) .

AANISH SINGLA

Reply

March 28, 2018 at 8:01 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152239)

IMO limitation of KNN comes into play when dimensions increase because in higher dimensions, finding neighbors which are quite close to each other in all dimensions might be tough, hence so called neighbors might be really far apart from each other which defeats the purpose of the algorithm.

Kindly share your thoughts/experiences.

AISHWARYA SINGH

Reply

March 29, 2018 at 3:07 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152261)

Hi Aanish,

Thank you for sharing your thoughts.

AMLESH KANEKAR

Reply

April 25, 2018 at 9:21 am (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152826)

I found it “inspiring”. Have spent last 4 months learning linear algebra, statistics, python. This learning list was culled from analyticsvidhya.com. Now just glimpsing through your article gives me the confidence to code knn from scratch. Thank you!

AISHWARYA SINGH

Reply

April 25, 2018 at 4:16 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152839)

Hi Amlesh,

Glad you found this useful!

AMLESH KANEKAR

Reply

May 2, 2018 at 6:46 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-152995)

I created my own dataset to experiment with KNN. When I plotted my data, the three targets/labels I have are extremely randomly distributed across the 2D plane … no clustering of the three colours is evident.
The Iris dataset shows a fairly high degree of clustering.
Should I continue with my dataset or there is the concept of "so-and-so distribution does not qualify for KNN"?
I can email a picture of my data plot if needed.

AMLESH KANEKAR

Reply

May 8, 2018 at 12:53 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-153109)

I figured this out. So it is fine if you do not respond.

MAX

Reply

October 6, 2018 at 12:21 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-155193)

That was very helpful. Thank you!
How to make the same visualization as in the pictures in section "How do we choose the factor K" ?

AISHWARYA SINGH

Reply

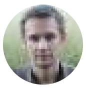
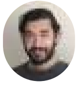
October 8, 2018 at 7:46 pm (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#comment-155234)

Hi Max,

For this, you will have to use a for loop. For each value of k, calculate the validation error and store in a separate list. Then plot these validation error values against k values.

TOP ANALYTICS VIDHYA USERS

Download Resource

Rank	Name		Points
1		SRK (https://datahack.analyticsvidhya.com/user/profile/SRK)	3988
2		Tezdhar (https://datahack.analyticsvidhya.com/user/profile/Tezdhar)	3986
3		mark12 (https://datahack.analyticsvidhya.com/user/profile/mark12)	3870
4		Denis Vorotyntsev (https://datahack.analyticsvidhya.com/user/profile/tearth)	3663
5		vopani Rao (https://datahack.analyticsvidhya.com/user/profile/Rohan_Rao)	3432

More User Rankings (https://datahack.analyticsvidhya.com/top-competitor/?utm_source=blog-navbar&utm_medium=web)

POPULAR POSTS

24 Ultimate Data Science Projects To Boost Your Knowledge and Skills (& can be accessed freely) (https://www.analyticsvidhya.com/blog/2018/05/24-ultimate-data-science-projects-to-boost-your-knowledge-and-skills/)

Essentials of Machine Learning Algorithms (with Python and R Codes) (https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/)

7 Types of Regression Techniques you should know! (https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression/)

Understanding Support Vector Machine algorithm from examples (along with code) (https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/)

A Complete Tutorial to Learn Data Science with Python from Scratch (https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-learn-data-science-python-scratch-2/)

Introduction to k-Nearest Neighbors: Simplified (with implementation in Python) (https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/)

A Simple Introduction to ANOVA (with applications in Excel) (https://www.analyticsvidhya.com/blog/2018/01/anova-analysis-of-variance/)

Download Resource

Stock Prices Prediction Using Machine Learning and Deep Learning Techniques (with Python codes) (https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-learningnd-deep-learning-techniques-python/)

RECENT POSTS

A Beginner's Guide to Hierarchical Clustering and how to Perform it in Python (https://www.analyticsvidhya.com/blog/2019/05/beginners-guide-hierarchical-clustering/)

MAY 27, 2019

Using the Power of Deep Learning for Cyber Security (Part 2) – Must-Read for All Data Scientists (https://www.analyticsvidhya.com/blog/2019/05/using-power-deep-learning-cyber-security-2/)

MAY 23, 2019

Data Science Project: Scraping YouTube Data using Python and Selenium to Classify Videos (https://www.analyticsvidhya.com/blog/2019/05/scraping-classifying-youtube-video-data-python-selenium/)

MAY 20, 2019

Statistics for Data Science: Introduction to t-test and its Different Types (with Implementation in R) (https://www.analyticsvidhya.com/blog/2019/05/statistics-t-test-introduction-r-implementation/)

MAY 16, 2019

ANALYTICS VIDHYA

About Us (http://www.analyticsvidhya.com/about-me/)

Our Team (https://www.analyticsvidhya.com/about-me/team/)

Career (https://www.analyticsvidhya.com/career-analytics-vidhya/)

Contact Us (https://www.analyticsvidhya.com/contact/)

Write for us (https://www.analyticsvidhya.com/about-me/write/)

DATA SCIENTISTS

Blog (https://www.analyticsvidhya.com/blog/)

Hackathon (https://datahack.analyticsvidhya.com/)

Discussions (https://discuss.analyticsvidhya.com/)

Apply Jobs (https://www.analyticsvidhya.com/jobs/)

Leaderboard (https://datahack.analyticsvidhya.com/users/)

COMPANIES

Post Jobs (https://www.analyticsvidhya.com/corporate/)

Trainings (https://trainings.analyticsvidhya.com)

Hiring Hackathons (https://datahack.analyticsvidhya.com/)

Advertising (https://www.analyticsvidhya.com/contact/)

Reach Us (https://www.analyticsvidhya.com/contact/)

JOIN OUR COMMUNITY :

f (https://www.facebook.com/Analytic

46336

(https://www.facebook.com/Analytic

Followers

(https://www.facebook.com/Analytic

(https://twitter.com/analytic

20765

(https://twitter.com/analytic

Followers

(https://twitter.com/analytic

G+ (https://plus.google.com/+Analyticsvi

(https://plus.google.com/+Analyticsvi

Followers

(https://plus.google.com/+Analyticsvi

in (https://in.linkedin.com/company/analytics-vidhya)

7513

(https://in.linkedin.com/company/analytics-vidhya)

Followers

(https://in.linkedin.com/company/analytics-vidhya)

Subscribe to emailer >

Privacy Policy (https://www.analyticsvidhya.com/privacy-policy/)

Terms of Use (https://www.analyticsvidhya.com/terms/)

Refund Policy (https://www.analyticsvidhya.com/refund-policy/)

Don't have an account? Sign up (h

Download Resource