بسمه تعالی

توضیحات پروژه داده کاوی و تحلیل نتایج

محمدرضا رضایی

941142208

Urmia University Of Technology

Information Mining From Data

Wireless Indoor Localization Data Set

**اطلاعات مربوط به دیتاست** :

Csv dataset file name : **wifi-localization.csv**

Number of features : **7**

Percent of test items : **20 %**

Imported lines : **2000**

Training items : **1600**

Test items : **400**

**Data Set Description**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Set Characteristics:** | Multivariate | **Number of Instances:** | 2000 | **Area:** | Computer |
| **Attribute Characteristics:** | Real | **Number of Attributes:** | 7 | **Date Donated** | 2017-12-04 |
| **Associated Tasks:** | Classification | **Missing Values?** | N/A | **Number of Web Hits:** | 36834 |

* **Abstract**:

Collected in indoor space by observing signal strengths of seven WiFi signals visible on a smartphone. The decision variable is one of the four rooms.

در فضای خانگی با مشاهده ی قوت و استحکام سیگینال از هفت سیگینال بیسیم قابل مشاهده روی گوشی های هوشمند جمع اوری شده است.

* **Data Set Information:**

Collected to perform experimentation on how wifi signal strengths can be used to determine one of the indoor locations.

جمع آوری شده برای انجام آزمایش در مورد چگونگی استفاده از قدرت سیگنال wifi برای تعیین یکی از مکان های داخلی.

* **Attribute Information:**

Each attribute is wifi signal strength observed on smartphone.

هر ویژگی ، قدرت سیگنال wifi است که در تلفن هوشمند مشاهده می شود.

لازم به ذکر است که **میزان** **داده** ی در نظر گرفته شده برای یادگیری در تمامی روش ها، **80** درصد و **20** درصد باقی جهت **تست** در نظر گرفته شده اند.

ما برای تحلیل این داده ها از روش های مختلف یادگیری ماشین، روش های مثل **NaiveBayes**، **KNN** و **SVM** و **Decision** **Tree** در پایتون استفاده کرده ایم تا تشخیص دهیم کدام یک از روش های ذکر شده بهترین روش برای یادگیری می باشد و بهترین نتایج را می دهد.

بهترین نتیجه، نتیجه ای خواهد بود **که کم ترین ارور و کمترین اختلاف** بین Training و Test را داشته باشد. که نشان دهنده ی این است که یادگیری به خوبی انجام شده است.

**در هر کدام از جدول ها بهینه ترین مقادیر با رنگ سبز مشخص شده است.**

در روش **SVM**، باید بهترین مقدار **برای C و Gamma** تعیین شود و بهترین روش این است که هر دو را همزمان باهم درنظر بگیریم یعنی به ازای مقادیر مختلف از هر دو بررسی انجام شود و بعد از اینکه GAMMA ی مناسبی پیدا کردیم به سراغ C میرویم.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C | GAMMA | Training  Accuracy | Test  Accuracy | Difference |
| 0.18 | 0.01 | 0.9875 | 0.9825 | 0.005 |
| 0.22 | 0.01 | 0.9875 | 0.985 | 0.0025 |
| 0.88 | 0.01 | 0.99063 | 0.9825 | 0.00812 |
| 0.09 | 0.01 | 0.98688 | 0.985 | 0.00187 |
| 0.458 | 0.01 | 0.99 | 0.9875 | 0.0025 |
| 0.1 | 0.01 | 0.98813 | 0.9775 | 0.1063 |
| 3.5 | 0.01 | 0.995 | 0.995 | 0 |
| 0.13 | 0.01 | 0.98688 | 0.985 | 0.00187 |
| 0.01 | 0.01 | 0.9775 | 0.9775 | 0 |
| 0.15 | 0.01 | 0.98688 | 0.985 | 0.00187 |
| 2.7 | 0.005 | 0.991875 | 0.9875 | 0.00437 |
| 1.5 | 0.001 | 0.984375 | 0.9875 | -0.00312 |
| 2.2 | 0.001 | 0.988125 | 0.98 | 0.00812 |
| 3 | 0.001 | 0.98687 | 0.98 | 0.00687 |
| 5 | 0.0001 | 0.983125 | 0.985 | -0.00187 |
| 10 | 0.00001 | 0.983125 | 0.975 | 0.00812 |

در جدول مشاهده میشود که پارامترهای مشخص شده با رنگ سبز هم روی دیتای ترین پرفورمنس خوبی داده اند هم روی داده ی تست بیشتر از ترین

در روش **Naïve Bayes** الگوریتمو چندبار اجرا میکنیم چون پارامتری نداره فقط احتمال هر کدام از دیتاهای ترینیگ و تست و اختلاف اونا بهترین حالت انتخاب میشه چون هر بار الگوریتم به صورت اتفاقی داده هارو مخلوط میکنه تو هر اجرا احتمالات متفاوتی خواهیم داشت.

|  |  |  |  |
| --- | --- | --- | --- |
| Run  Round | Training  Accuracy | Test Accuracy | difference |
| 1 | 0.985 | 0.9775 | 0.0074 |
| 2 | 0.98975 | 0.99 | -0.00025 |
| 3 | 0.983125 | 0.985 | -0.00187 |
| 4 | 0.9825 | 0.9875 | -0.005 |
| 5 | 0.985 | 0.985 | 0 |
| 6 | 0.98375 | 0.9825 | 0.00125 |
| 7 | 0.98625 | 0.9775 | 0.00874 |
| 8 | 0.985625 | 0.9725 | 0.01312 |
| 9 | 0.981875 | 0.99 | -0.00812 |
| 10 | 0.984375 | 0.985 | -0.00062 |

در این اجرا که با رنگ سبز مشخص شده کمترین اختلاف و دقت تست نیز بیشتر از ترین هستش.

در روش **KNN**، متغییر K رو باید تغییر بدیم اگر کم بدیم احتمال overfitting زیاده و اگر K رو زیاد بدیم underfitting رخ میده. بعد از بررسی های پی در پی K رو به صورت زیر تغییر دادیم:

|  |  |  |  |
| --- | --- | --- | --- |
| K | Training Accuracy | Test Accuracy | difference |
| 2 | 0.991875 | 0.98 | 0.01187 |
| 3 | 0.98938 | 0.9875 | 0.00187 |
| 7 | 0.985 | 0.9825 | 0.0025 |
| 6 | 0.99 | 0.9835 | 0.0065 |
| 8 | 0.98625 | 0.9825 | 0.00375 |
| 10 | 0.98563 | 0.98 | 0.00562 |
| 9 | 0.98313 | 0.98 | 0.00313 |
| 5 | 0.988125 | 0.985 | 0.00312 |
| 12 | 0.98125 | 0.99 | -0.000875 |
| 15 | 0.983125 | 0.97 | 0.1312 |
| 17 | 0.983125 | 0.9625 | 0.02062 |
| 20 | 0.9825 | 0.9825 | 0 |
| 4 | 0.993125 | 0.985 | 0.00812 |

طبق جدول بالا : 12 = K

در روش **Decision Tree** ، عمق درخت رو تغییر میدیم. اگر مقدار k رو خیلی بدیم overfitting رخ میده و اگر مقدارشو کم بدیم underfitting رخ میده. مقدار عددی K رو هم به صورت زیر تغییر دادیم:

|  |  |  |  |
| --- | --- | --- | --- |
| Max-depth | Training Accuracy | Test Accuracy | difference |
| 1 | 0.50125 | 0.495 | 0.00625 |
| 3 | 0.97313 | 0.9725 | 0.00062 |
| 2 | 0.74875 | 0.7425 | 0.00625 |
| 5 | 0.98438 | 0.9775 | 0.00687 |
| 6 | 0.986875 | 0.9775 | 0.00937 |
| 8 | 0.99375 | 0.9775 | 0.01624 |
| 10 | 0.993125 | 0.9875 | 0.00562 |
| 15 | 1.0 | 0.96 | 0.04 |
| 20 | 1.0 | 0.9775 | 0.02249 |

طبق جدول بالا : 3 = Max-depth

جدول زیر به نتایج کلی هر روش پس از چندین بار اجرای هر کد و رسیدن به حداکثر بازدهی را نمایش می دهد.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Training Accuracy | Test Accuracy | difference |
| SVM | 0.984375 | 0.9875 | -0.00312 |
| NB | 0.98975 | 0.99 | -0.00025 |
| KNN | 0.98125 | 0.99 | -0.000875 |
| DT | 0.97313 | 0.9725 | 0.00062 |

طبق جدول بالا روش NaiveBayes از بقیه روش ها بهینه تر می باشد.