Mini Use-Case 15: Random Forest für das klassifizieren von Bildern

Wie auch schon in den vorherigen Use-Cases logistische Regression (11), SVM (12) und k-NN (13) wurde die Fragestellung bearbeitet, wie mittels KI ein Arbeitsplatz sicherer gestaltet werden kann. Drei verschiedene Modelle wurden eingesetzt, um einen Arbeitsplatz auf potenziell gefährliche Werkzeuge zu überwachen. Der vorliegende Use-Case ist der letzte Use-Case, bei dem selbe Fragestellung mit einem vierten neuen Modell bearbeitet wird. In diesem Use-Case wir das sogenannte Random Forest Modell eingesetzt. Erneut sind die Datenaufbereitung und Klassifizierung im selben Stil aufgebaut. Mehr Informationen dazu sind erneut im Use-Case 11 logistische Regression genau beschrieben. Abbildung 1 dient zur Erinnerung der Datenaufbereitung und wie der Input für die Random Forest Klassifizierung aussieht.

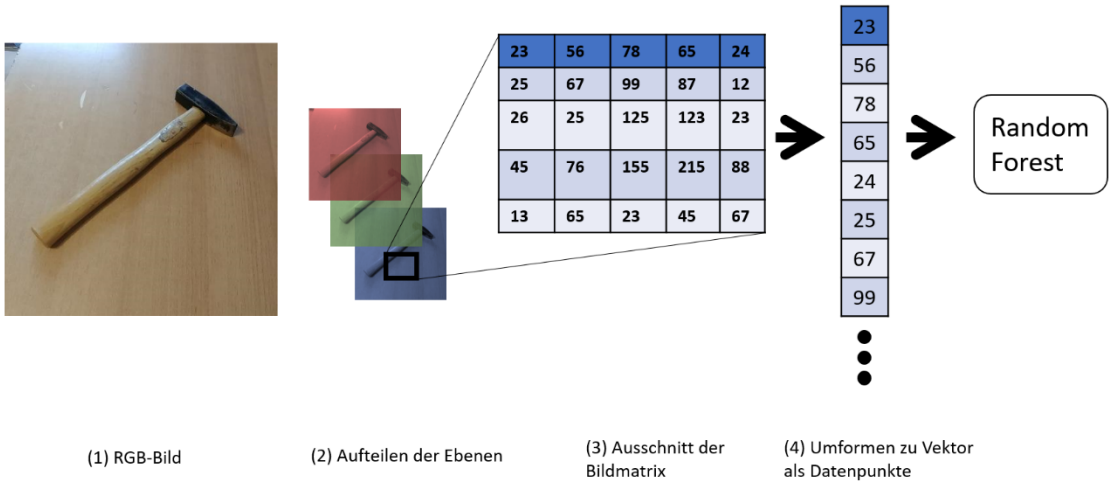

Abbildung 1: Aufbereiten der Daten ((1) RGB-Bild welches in einen Spaltenvektor umgewandelt werden soll, (2) Aufteilung der Farbebenen in Rot, Grün und Blau, (3) Visualisierung eines Teiles der Bildmatrix, (4) Darstellung des Spaltenvektors im Vergleich zur Bildmatrix und Input für den Random Forest)

Wie bereits angesprochen wird bei diesem Use-Case ein Random Forst („Zufallswald“ oder auch „zufälliger Wald“) eingesetzt. Dieses Modell basiert auf der Verwendung mehrerer Entscheidungsbäume und kann die eben besprochenen Bild-Vektoren als Datenpunkte verwenden, um eine Entscheidung zu treffen. Was sind aber Entscheidungsbäume und wie treffen diese eine Entscheidung?

Ein Entscheidungsbau betrachtet eine gewisse Fragestellung und kategorisiert damit den vorliegenden Datenpunkt. Das Ziel ist es genug „Fragen“ zu stellen, um eine Entscheidung treffen zu können die nur einer Klasse entspricht. In Abbildung 2 sind vier verschiedene Klassen eingezeichnet, die jeweils über einen X und Y Wert verfügen. In unserem Beispiel könnten die zweidimensionalen Punkte mit der PCA vorverarbeitete Bilder sein. Wird nun ein neuer Datenpunkt klassifiziert müssen ausreichend Fragen gestellt werden, um herauszufinden „wo dieser am besten dazu passt“.

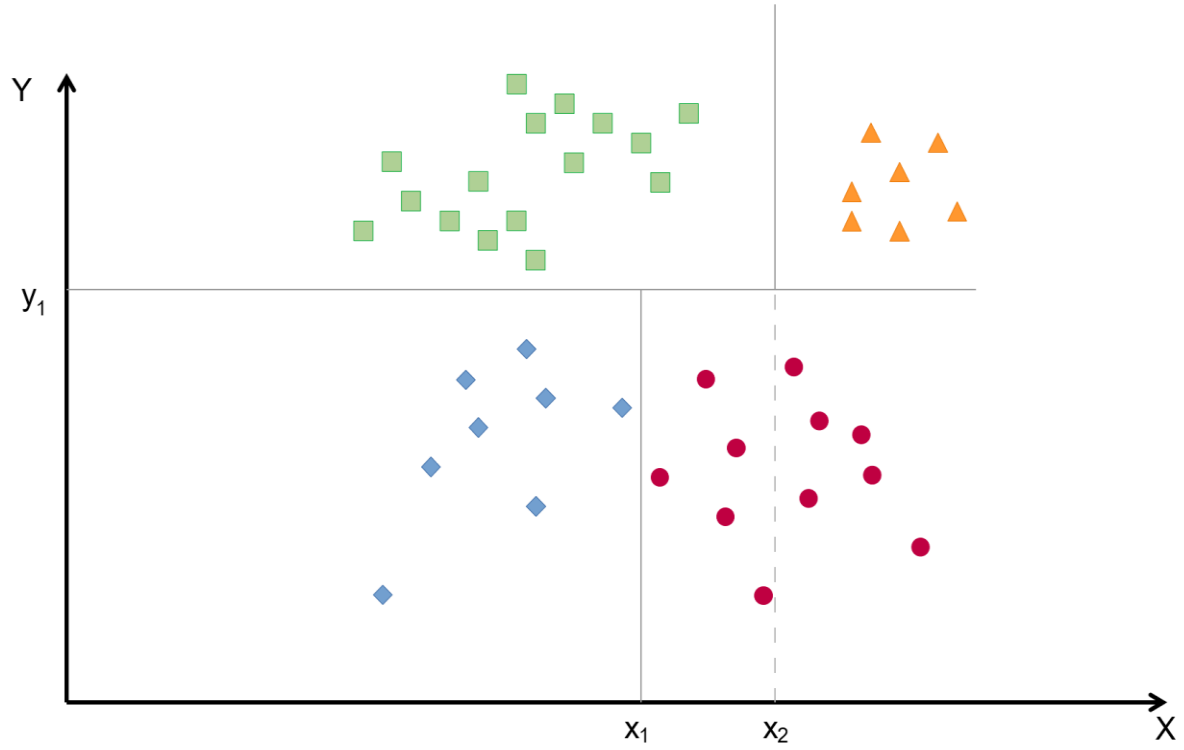

Abbildung 2:Beispieldatensatz für einen Entscheidungsbaum

Ein neuer Datenpunkt verfügt Beispielsweiße über einen y-Wert größer dem Grenzwert y_1 und einem x-Wert größer als x_1 aber kleiner als x_2. Zu welcher Klasse gehört nun dieser neue Datenpunkt? Dies kann über einen Entscheidungsbaum herausgefunden werden. Der Random Forrest sucht sich selbstständig zufällige Beobachtungspunkte und Merkmale für die verschiedenen Bäume aus und berechnet den durchschnitt der Ergebnisse. So können individuelle Fehler einzelner Bäume über die Gesamtmenge (dem Wald) ausgeglichen werden. [1]
Eine möglicher Entscheidungsbaum ist in Abbildung 3 dargestellt. Die Entscheidungen werden von unten nach oben getroffen. Jede Gabel beschreibt ein mögliches Ergebnis. So bauen sich die „Äste“ des Baumes von unten nach oben auf. Als ersten Schritt wird hier abgefragt, ob der neue x-Wert kleiner als x_1 ist. Dies trifft in dem vorliegenden Beispiel nicht zu. Der rote Pfeil gibt an, dass die Fragestellung nichtzutreffend ist, weshalb die nächste Gabelung abfragt, ob der y-Wert kleiner als der Grenzwert y_1 ist. Die bisher gestellten Fragen grenzen, das Ergebnis schon ein allerdings könnte es sich immer noch um das grüne Rechteck oder das orangene Dreieck handeln. Dementsprechend wird eine neue Frage gestellt, welche eine definierte Antwort liefert. Bei dem Beispiel ist der x-Wert kleiner als x_2, weshalb der neue Datenpunkt zu der Klasse „grünes Rechteck“ gehören muss [2].

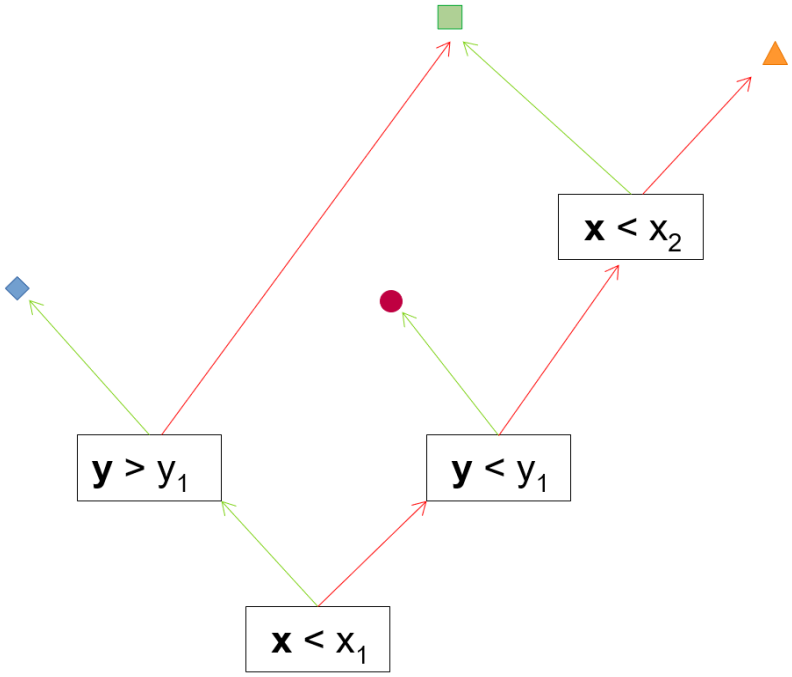

Abbildung 3: Entscheidungsbaum anhand der Beispieldaten. Pfeile (Legende): Grün = Aussage ist wahr, Rot = Aussage stimmt nicht

Wie der Name schon vorschlägt, besteht der Random Forest aus mehreren Bäumen – um genau zu sein aus den gerade besprochenen Entscheidungsbäumen. Das Random Forest Modell entscheidet darüber wie viele Entscheidungsbäume generiert werden sollen. Jeder einzelne generierte Baum wird anschließend für die Gesamtentscheidung herangezogen. Der Vorteil im Gegensatz zu anderen Modellen ist, dass es sich nicht um eine „Black-Box“ handelt. Es ist recht einfach – zumindest bei einem einzelnen Entscheidungsbaum - die Entscheidungen zu Visualisieren und zu verfolgen. In unserem Fall bedeutet das eine erklärbare Trennung der mit der PCA vorverarbeiteten und reduzierten Vektoren. Wie dies auch in Abbildung 3 dargestellt ist. Ebenso lassen sich die einzelnen Bäume gut parallelisieren, was für einen sehr schnellen Trainingsprozess sorgt [3].

Die Überwachung des Arbeitsplatzes mittels eines Random Forrest Modelles – welches mittels der sklearn-Library implementiert wurde [4] – funktioniert mäßig gut. Im Vergleich zu den anderen Use-Cases dieser Art (11, 12 und 13) ist es dieses Modell, der die schlechteste Klassifizierung aufweist. Optimierungen können wie auch schon in den anderen drei Use-Cases durchgeführt werden (Augmentation, Qualität des Datensatzes oder komplexeres Modell wählen). Mehr Informationen dazu stehen im Use-Case 11 logistische Regression. Ebenso hat – wie auch bei den anderen Use-Cases dieser Art – eine Umwandlung in ein Graustufenbild zu keiner Verbesserung geführt.

Bevor das Modell trainiert wurde, muss es definiert werden. Hierzu wurde die maximale suchtiefe auf 5 Ebenen gesetzt sowie die Anzahl der Bäume auf 10. Diese Werte haben sich durch Testen und Ausprobieren ergeben. Zum Evaluieren des Modells wurde wie auch bei den anderen Use-Cases ein

Trainings- und Test-Datensatz eingesetzt. Das bedeutet, dass in unserem Fall 70% aller 954 Bilder verwendet, werden um das Model zu Trainieren. Mit den restlichen 30% wird das Model getestet, um zu schauen, wie gut das Modell funktioniert. Der Accuracy Score belief sich hierbei auf 0.92 (von 1.0).

Dieses Modell ist allerdings ein sehr guter Einstieg in das Thema KI/AI. Wie schon angesprochen, ist es keine Black-Box und auch einfach zu implementieren.

[1] Donges, Niklas, „builtin.com“, 2021, [Online]. Available: https://builtin.com/data-science/random-forest-algorithm. [Zugegriffen am 09.11.2021]

[2] Kossen J., Müller M.E., Ruckriegel M. (2019) Entscheidungsbäume. In: Kersting K., Lampert C., Rothkopf C. (eds) Wie Maschinen lernen. Springer, Wiesbaden. https://doi-1org-10003429b00f4.han.technikum-wien.at/10.1007/978-3-658-26763-6_15

[3] Aunkofer B. (2017) „data-science-blog.com“, 2021. [Online]. Available: https://data-science-blog.com/blog/2017/02/13/entscheidungsbaumverfahren-artikelserie/ [Zugriff am 13 September 2021].

[4] scikit-learn developers, „scikit-learn.org“, 2021. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html. [Zugriff am 5 Oktober 2021].