Computerlinguistik II: Übung

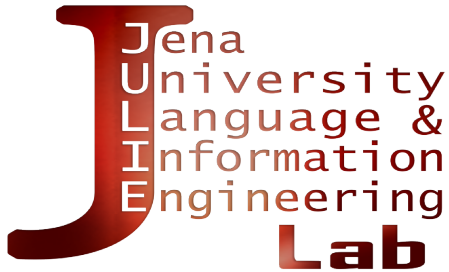
Maschinelles Lernen

Sven Büchel

Jena Language & Information Engineering (JULIE) Lab
Friedrich-Schiller-Universität Jena, Germany

https://julielab.de/

Sommersemester 2019



Basiskonzepte aus der Linearen Algebra

Kurzeinführung ins Maschinelle Lernen

Abschnitt 1

Basiskonzepte aus der Linearen Algebra

Vektoren

Ein Vektor ist eine Zusammenfassung mehrerer, nicht notwendigerweise verschiedener Elemente, wobei die Reihenfolge der Elemente eine Rolle spielt. Wir betrachten hier Vektoren aus $\mathbb{R}^n$, $n \in \mathbb{N}_+$.
Zum Beispiel

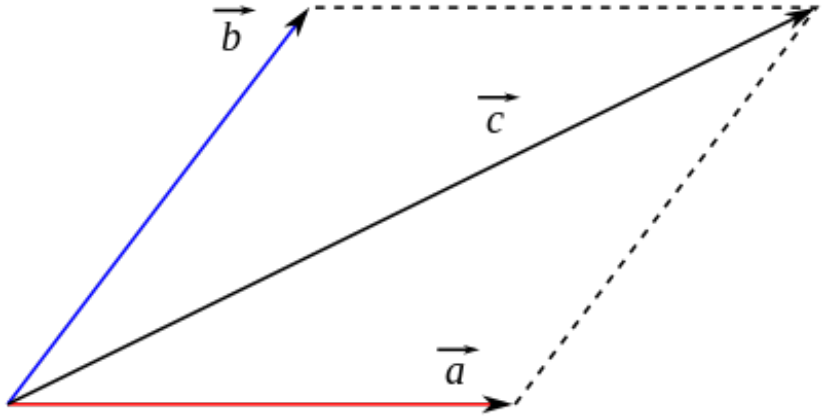
$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = (x_1, x_2, \ldots, x_n)$$

mit $x_1, \ldots x_n \in \mathbb{R}$.

Einfache Vektoroperationen

Seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ Vektoren und $c \in \mathbb{R}$ ein Skalar.
Vektoraddition:

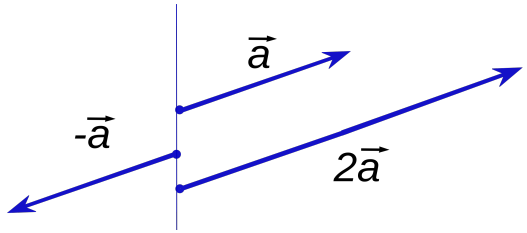
$$\mathbf{a} + \mathbf{b} = \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_n + b_n \end{pmatrix}$$



Bildquelle: https://de.wikipedia.org/wiki/Vektor

Skalarmultiplikation:

$$c \cdot \mathbf{a} = \begin{pmatrix} c \cdot a_1 \\ c \cdot a_2 \\ \vdots \\ c \cdot a_n \end{pmatrix}$$



Bildquelle: https://de.wikipedia.org/wiki/Vektor

Skalarprodukt

Seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ Vektoren. Das **Skalarprodukt** (auch **Inneres Produkt**) der beiden Vektoren ist

$$\mathbf{a} \cdot \mathbf{b} := \sum_{i=1}^{n} a_i \cdot b_i = a_1 b_1 + a_2 b_2 + \ldots a_n b_n$$

Es ist abhängig von der Länge der Vektoren sowie dem Winkel zwischen beiden (siehe nächste Folien).

Länge von Vektoren

Sei $\mathbf{a} \in \mathbb{R}^n$. Die Länge (auch der Betrag) von $\mathbf{a}$ (euklidische Norm) beträgt

$$|\mathbf{a}| := \sqrt{\sum_{i=1}^{n} {a_i}^2}$$

Winkel und Kosinusähnlichkeit zwischen Vektoren

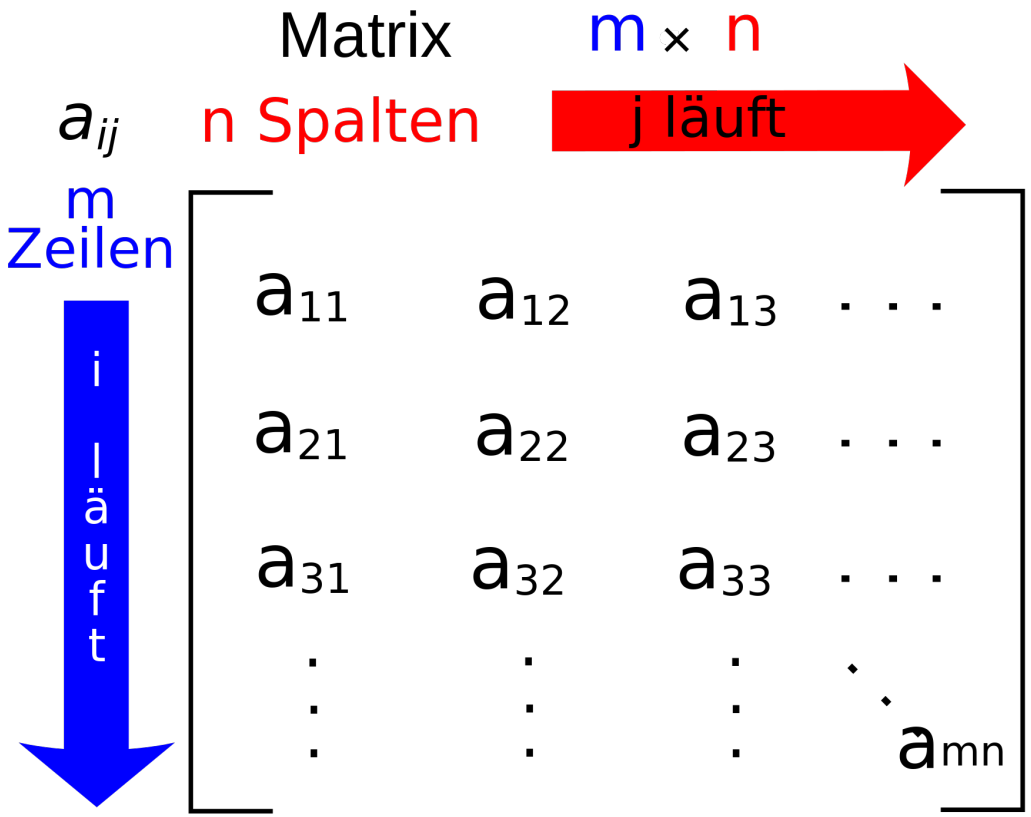
Seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ Vektoren. Der Winkel zwischen Vektoren lässt sich über den Zusammenhang

$$\cos(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}||\mathbf{b}|}$$

berechnen, wobei $\cos(\mathbf{a}, \mathbf{b})$ der Kosinus zwischen den beiden Vektoren bezeichnet.
In der Computerlinguistik werden linguistische Einheiten (Wörter, Sätze, Texte, usw.) häufig als Vektoren dargestellt, die mathematisch auf ihre Ähnlichkeit hin verglichen werden. Dies geschieht i.d.R. nicht über den Winkel. Stattdessen wird direkt der Kosinus des Winkels (liegt zwischen -1 und 1) als Ähnlichkeitsmaß verwendet (**Kosinusähnlichkeit**).

Matrizen

Eine Matrix ist eine rechteckige Anordnung von Elementen (hier wieder aus $\mathbb{R}$).



Notation von Matrizen

Sei $A \in \mathbb{R}^{m \times n}$ eine $m \times n$-Matrix. Diese kann dargestellt werden als

$$A = (a_{ij})_{i=1,\ldots,m;j=1,\ldots,n} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix}$$

Einfache Matrixoperationen

Seien $A, B \in \mathbb{R}^{m \times n}$ Matrizen und sei $c \in \mathbb{R}$ ein Skalar.
Matrizenaddition:

$$A + B := (a_{ij} + b_{ij})_{i=1,\ldots,m;j=1,\ldots,n}$$
$$= \begin{pmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \ldots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & \ldots & a_{2n} + b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \ldots & a_{mn} + b_{mn} \end{pmatrix}$$

Skalarmultiplikation:

$$c \cdot A := (c \cdot a_{ij})_{i=1,\ldots,m;j=1,\ldots,n}$$
$$= \begin{pmatrix} ca_{11} & ca_{12} & \ldots & ca_{1n} \\ ca_{21} & ca_{22} & \ldots & ca_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ ca_{m1} & ca_{m2} & \ldots & ca_{mn} \end{pmatrix}$$

Matrizenmultiplikation

Sei A eine $l \times m$-Matrix und B eine $m \times n$-Matrix. Das Produkt $A \cdot B$ ist eine $l \times n$-Matrix C mit

$$c_{ij} := \sum_{k=1}^{m} a_{ik} \cdot b_{kj}$$

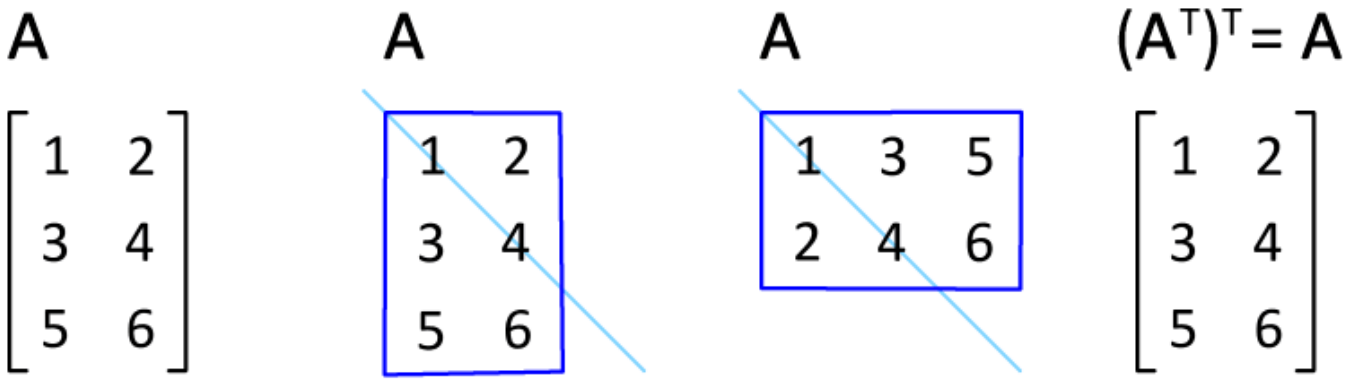
Zur Veranschaulichung und händischen Berechnung kann das **Falk-Schema** verwendet werden.

			Spalte j	
			1	*2*
			–1	**1**
Zeile i			**1**	**–2**
1	**1**	**4**	**3**	**–7**
2	**2**	**5**	**3**	**–8**
3	**3**	**–6**	**–9**	**15**

https://de.wikipedia.org/wiki/Falksches_Schema

Transponierte Matrix

Die **Transponierte einer** $m \times n$**-Matrix** $A = (a_{ij})$ ist die $n \times m$-Matrix $A^T = (a_{ji})$.



https://de.wikipedia.org/wiki/Matrix_(Mathematik)

Vektoren und Matrizen

- Ein Vektor mit n Elementen lässt sich auch als $n \times 1$-Matrix aufassen.
- Daher lässt sich auch zu einem Vektor die Transponierte berechnen, die dann eine $1 \times n$-Matrix ergibt (Spaltenvektor vs. Zeilenvektor).
- Das Skalarprodukt von zwei Vektoren $\mathbf{a} \cdot \mathbf{b}$ kann somit auch als Matrizenmultiplikation aufgefasst werden:

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b}$$

Abschnitt 2

Kurzeinführung ins Maschinelle Lernen

Anwendungsszenario 1

Sie betreiben bereits seit vielen Jahren eine Fluggesellschaft. Ein wesentlicher Faktor die Flugpreise niedrig zu halten und die Umwelt zu schonen besteht darin, die Flugzeuge nicht unnötig stark zu betanken, da das überflüssige Kerosin lediglich als Balast wirkt. Sie versuchen zu ermitteln, wie viel Treibstoff für die nächsten Flüge mitgeführt werden muss. Sie können sich dabei auf die entsprechenden Betriebsstatistiken aus zahlreichen Geschäftsjahre stützen, müssen jedoch jeweils die Passagierzahl, das Frachtgewicht, sowie die aktuelle Wettervorhersage miteinbeziehen.

Anwendungsszenario 2

Sie betreiben eine nationale Postgesellschaft. Bisher wurden Briefe dadurch zugestellt, indem Mitarbeiter die handschriftlichen Adressangaben abgetippt haben und in das Postcomputersystem eingegeben haben. Sie stellen jedoch fest, dass dieser Schritt sowohl einen Großteil der Kosten, als auch der Zustellungsfehler verursacht. Daher würden Sie die handschriftlichen Zustelladressen gerne automatisch erkennen.

Anwendungsszenario 3

Sie betreiben eine Universitätsbibliothek. Da in den letzten Jahren ein Großteil Ihres Katalogs ausgetauscht wurde, möchten Sie die Gelegenheit nutzen und ein neues Kategoriensystem für ihre Bücher einrichten. Sie haben jedoch nicht genügend Experten zur Verfügung, um für jedes Fachgebiet die “korrekten” Untergebiete zu ermittlen. Gibt es einen Weg diese automatisch aus den Ähnlichkeitsbeziehungen der Bücher untereinander abzuleiten?

Anwendungsszenario 4

Sie bekommen leider viel zu viele Spam-Emails. Wenn es nur eine Möglichkeit gäbe, diese automatisch auszusortieren!

Wichtige Teilgebiete des Maschinellen Lernens

- **Überwachtes Lernen**
  - Die Instanzen des Trainingsdatensatzes bestehen aus Merkmalen (Features, Eingabevariable, unabhängige Variablen) sowie Labels (Ziel/Ausgabevariablen, abhängige Variablen)
  - Ziel ist es aus den Trainingsdaten ein Modell abzuleiten, das neuen, ungesehenen Instanzen möglichst gute Labels zuordnet.
- **Unüberwachtes Lernen**
  - Es gibt keine Labels. Stattdessen soll die Struktur der eingegebenen Daten gelernt werden. Welche Gruppen (Cluster) ergeben sich aus den Daten? Sind alle Merkmale zur Beschreibung der Daten notwendig oder können manche auch weggelassen oder zusammengefasst werden (Dimensionsreduktion).
- **Halbüberwachtes Lernen**
  - Mache Instanzen der Trainingsdaten tragen Labels aber nicht alle. Das Ziel ist das selbe wie beim überwachten Lernen. Jedoch bieten ungelabelte Instanzen zusätzliche Hilfestellung.

Grundstruktur des Überwachten Lernens

- Sei $T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ eine Menge von Merkmal-Label-Paaren.
- Das Ziel ist eine Funktion f (Modell) aus T abzuleiten, sodass für alle Instanzen i $f(x_i) = \hat{y}_i$ möglichst genau y_i approximiert und dabei gleichzeitig möglichst gut auf ungesehene Daten generalisiert.
- D.h. das Modell soll die zugrundeliegende Zusammenhänge zwischen Merkmalen und Labels lernen, anstatt die Trainingsbeispiele “auswendig zu lernen”.

Grundstruktur des Überwachten Lernens II

- Statt dieser mengenbasierten Notation wird in Praxis meist auf Vektoren und Matrizen zurückgegriffen.
- Die Merkmale der Instanz i werden durch den Merkmalsvektor $\mathbf{x}_i$ repräsentiert. Alle Merkmale aller Instanzen werden in der Merkmalsmatrix $X \in \mathbb{R}^{m \times n}$ zusammengefasst (Zeile bezeichnen Instanzen, Spalten Merkmale).
- Die Labels werden ebenfalls in der Labelmatrix Y zusammengefasst. Bei nur einer Ausgabevariable ist dies ein $m \times 1$ Spaltenvektor.

“Lebensphasen” eines ML-Modells

1. **Training**. Ein ML-Algorithmus wird genutzt um das Model aus den Daten zu erstellen.
2. **Test**. Das fertige Modell wird auf Grundlage von bisher zurückgehaltenen Daten hinsichtlich einer bestimmten Metrik (z.B. Korrelation, Fehler, Genauigkeit) evaluiert. Hierfür sind *neue* Mermal-Label-Paare nötig.
3. **Inferenz**. Neue Merkmalsvektoren werden in das fertige Modell eingespeist woraufhin dieses Vorhersagen bzgl. der Label macht. Kenntnis der korrekten Label wird *nicht* mehr benötigt.

Messniveaus

Messniveaus (auch Skalenniveaus) unterteilen Messwerte (Daten) danach, welche logische und mathematischen Operationen sich sinnvoll darauf anwenden lassen. Auf Daten eines höheren Messniveaus können dabei auch Operationen eines niedrigeren angewandt werden.

Name	**Sinnvolle Operationen**	**Lagemaß**	**Beispiel**
Nominal (Kategorial)	gleich/ungleich	Modus	Geschlecht
Ordinal	größer/kleiner	Median	Bildungsgrad
Metrisch (Numerisch)	Addition/Multiplikation	Mittelwert	Sprecheralter

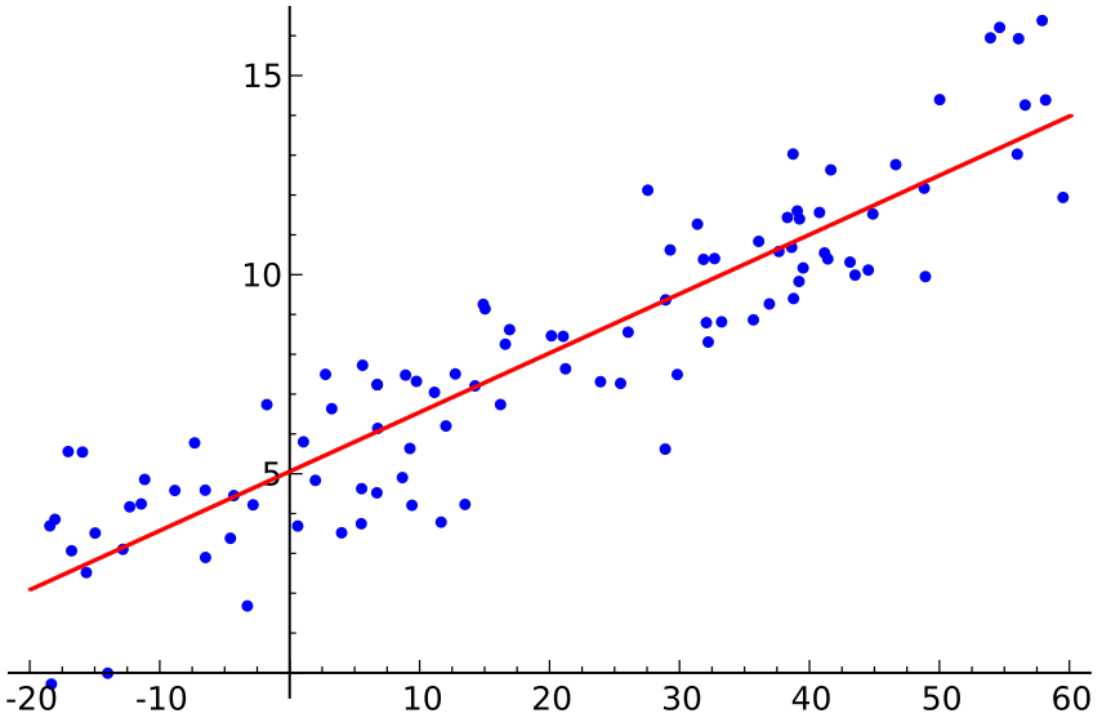
Klassifikation vs. Regression

- ML-Probleme mit nominalen Zielvariblen heißen **Klassifikation**, solche mit numerischen Zielvariablen **Regression**.
- Manche Arten von Modellen sind sind lediglich für eine beiden Problemarten anwendbar, andere für beide.
- Bei Regressionsproblemen bestehen die Einträge der Labelmatrix aus reellen Zahlen, bei Klassifikationsproblemen ist es etwas komplizierter (siehe nächste Folie).

Kodierung der Labels bei Klassifikationsproblemen

- Ein Klassifikationsproblem heißt **binär**, wenn die Zielvariable genau zwei Werte einnehmen kann (z.B. *Spam* oder *kein Spam*). Bei binären Klassifikationsproblemen wird die Zielvariable häufig mit 0 und 1 oder -1 und 1 kodiert.
- Ein **Multiklassenproblem** ist ein Klassifikationsproblem, bei dem die Zielvariable mehr als zwei Werte einnehmen kann (z.B. Augenfarbe oder Wortart). Hierbei wird das Label meist mit binären Dummy-Variablen kodiert. Bei n möglichen Klassen, besteht jeder Labelvektor aus n Elementen, die jeweils 0 oder 1 sein können.

Lineare Regression (Intuition)



https://de.wikipedia.org/wiki/Einfache_lineare_Regression

Die Vorhersagen des LR-Models liegen auf einer Geraden (eine Merkmalsvariable), einer Ebene (zwei Merkmalsvariablen), oder einer Hyperebene (noch mehr Merkmalsvariablen).

Lineare Regression (Formal)

Ein Lineares Regressionsmodell erzeugt Vorhersagen durch

$$\hat{y} = f(\mathbf{x}) = \beta_0 + \sum_{i=0}^{n} \beta_i x_i = \beta_0 + \beta \cdot \mathbf{x}$$

mit $\beta = (\beta_1, \beta_2, \ldots, \beta_n)$.

- β_0 nennt sich Bias oder Intercept, $\beta_1, \beta_2, \ldots, \beta_n$ nennen sich Gewichte oder Koeffizienten. Zusammen heißen sie (Model-) Parameter.
- Die Parameter lassen sich effektiv durch eine mathematische Formel ermitteln (wird in der Übung nicht behandelt).

Effektive Inferenz mit LR

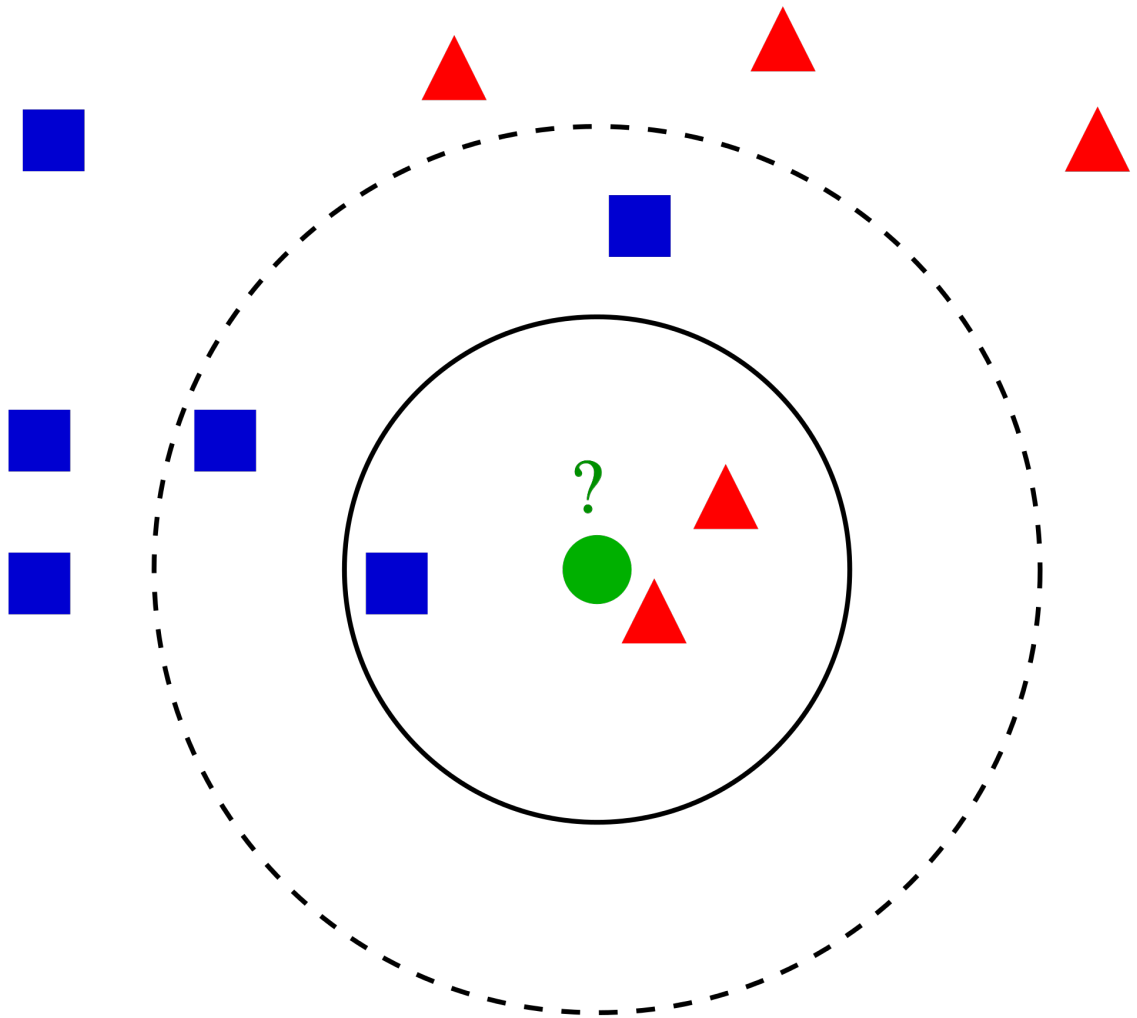
Ein einmal erzeugtes LR-Modell kann sehr effektiv große Mengen neuer Instanzen labeln. Sei X' eine neue Merkmalsmatrix, dann ist

$$\hat{Y}' = X' \cdot \beta + (\beta_0, \beta_0, \ldots, \beta_0)$$

K-Nearest-Neighbor (kNN)

- kNN (*k* nächste Nachbarn) ist einfacher Algorithmus für Regression *und* Klassifikation.
- Vorraussetzung ist ein Abstands- oder Ähnlichkeitsmaß, mit dem beurteilt werden kann, welches für eine gegebene Instanz die *k* nächsten Nachbarn im Merkmalsraum sind (z.B. Kosinusähnlichkeit oder Euklidischer Abstand).
- Training: Findet nicht statt. Das Modell "merkt" sich stattdessen alle Instanzen des Trainingsdatensatzes.
- Inferenz:
  - Klassifikation: Wähle die Klasse, die bei den *k* nächsten Nachbarn am häufigsten Auftritt.
  - Regression: Bilde den Mittelwert der *k* nächsten Nachbarn.

Illustration kNN



https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

Weisen Sie den Einstiegsbeispielen jeweils eines der drei Teilgebiete (überwachtes, unüberwachtes, halbüberwachtes Lernen) zu. Begründen Sie ihre Antwort.