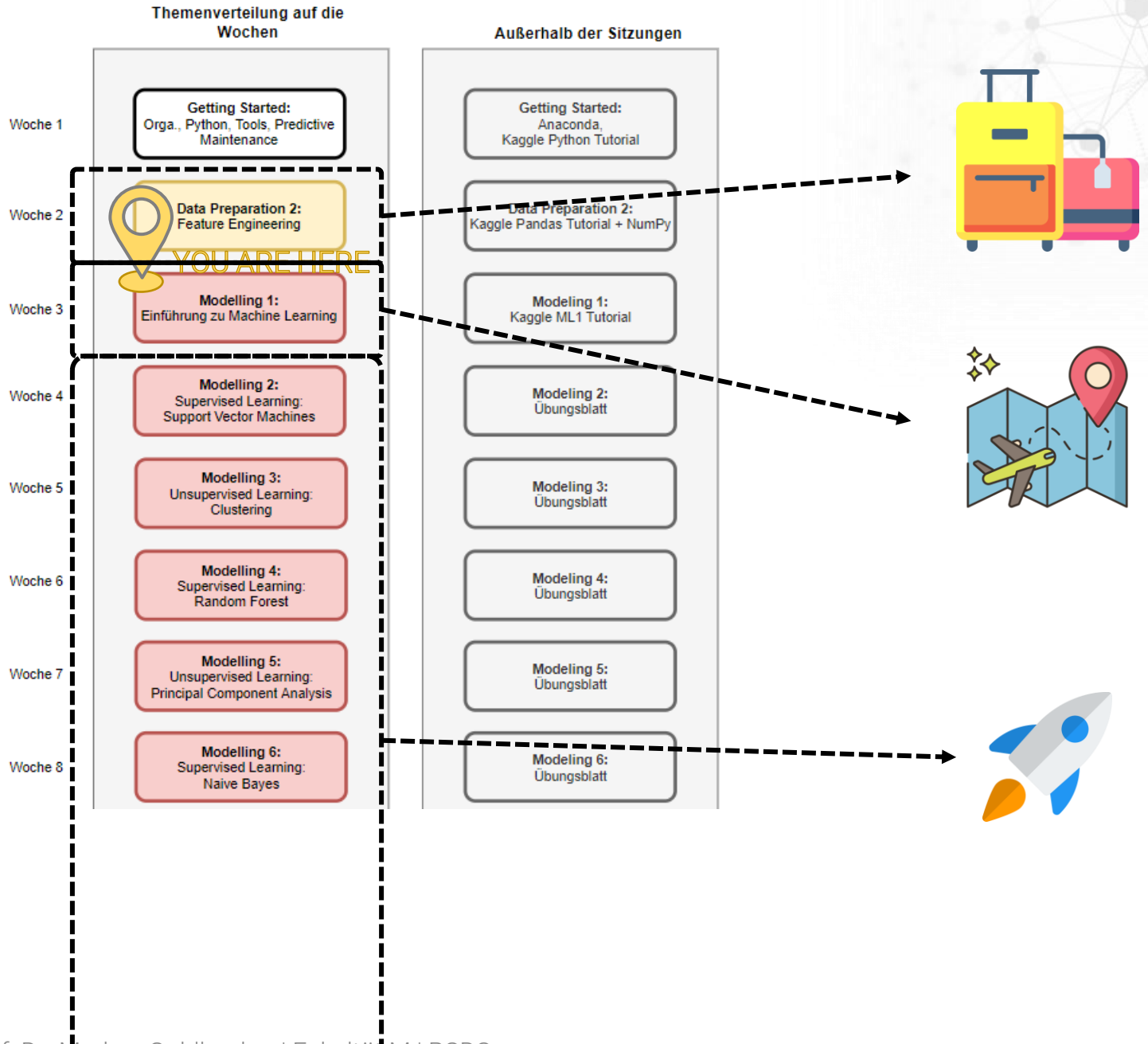
Modeling

Einführung in Machine Learning

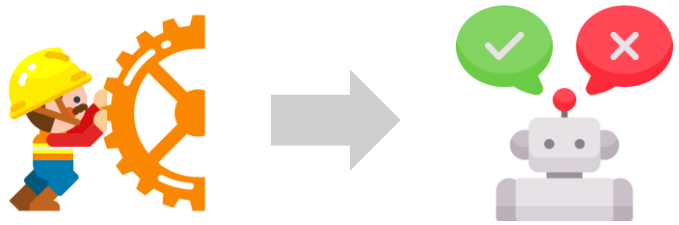
Agenda

1. Was ist Machine Learning?
2. Kategorien des Machine Learning und qualitative Beispiele
3. Machine Learning in Python: `scikit-learn`
4. Hyperparameter und Model Validation
5. Bias-Variance Trade-off
6. Cross-Validation und Generalisierung
7. Kostenfunktionen
8. Machine Learnining Pipelines (evtl. in der letzten Sitzung bzw. extra Foliensatz erstellen)

Wo sind wir?


Themenverteilung auf die Wochen
Außerhalb der Sitzungen
Woche 1
Getting Started: Orga., Python, Tools, Predictive Maintenance
Getting Started: Anaconda, Kaggle Python Tutorial
Woche 2
Data Preparation 2: Feature Engineering
YOU ARE HERE
Data Preparation 2: Kaggle Pandas Tutorial + NumPy
Woche 3
Modelling 1: Einführung zu Machine Learning
Modeling 1: Kaggle ML1 Tutorial
Woche 4
Modelling 2: Supervised Learning: Support Vector Machines
Modeling 2: Übungsblatt
Woche 5
Modelling 3: Unsupervised Learning: Clustering
Modeling 3: Übungsblatt
Woche 6
Modelling 4: Supervised Learning: Random Forest
Modeling 4: Übungsblatt
Woche 7
Modelling 5: Unsupervised Learning: Principal Component Analysis
Modeling 5: Übungsblatt
Woche 8
Modelling 6: Supervised Learning: Naive Bayes
Modeling 6: Übungsblatt

Ausgangssituation



Wo befinden wir uns jetzt?

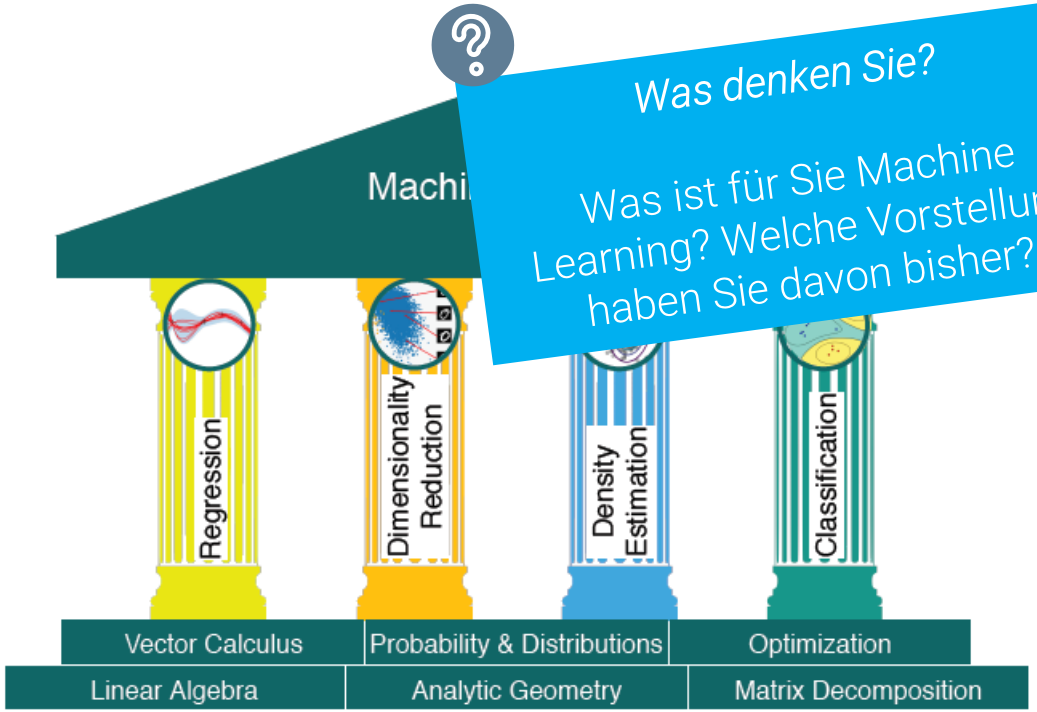
- Wir wissen jetzt was ein Feature ist
- Wir haben eine Vorstellung von einem Feature-Raum
- Wir wissen, wie Daten in einem Feature-Raum repräsentiert werden
- Wir wissen wie man Daten exploriert und **„manuell“** Schlüsse zieht (deskriptiv und inferenzstatistisch)

→ Nun wollen wir, dass Algorithmen für uns Schlüsse ziehen und Urteile fällen!

So what?

Interessanter Aspekt vorab: im Machine Learning bringen wir Algorithmen bei *wissenschaftlich zu Denken bzw. zu arbeiten*

Was ist Machine Learning?



Aus Deisenroth et al. „Mathematics for Machine Learning“

Was denken Sie?

Was ist für Sie Machine Learning? Welche Vorstellung haben Sie davon bisher?

Machine Learning == „Building Models of Data“

- Machine Learning zeichnet sich dadurch aus, dass mathematische Modelle entstehen, um zugrundeliegende **Daten** zu **verstehen**
- Diese Modelle werden an die **Daten angepasst**, damit sie diese so optimal wie möglich repräsentieren
- Diese Anpassung kann stattfinden, da Machine Learning Modelle „**tunable parameters**“ haben
  - → Anpassung dieser auf die Daten
  - → Das Modell **lernt** aus den Daten
- Nach diesem Lernprozess können trainierte Modelle dazu verwendet werden, um **Labels** ungesehener Daten vorherzusagen

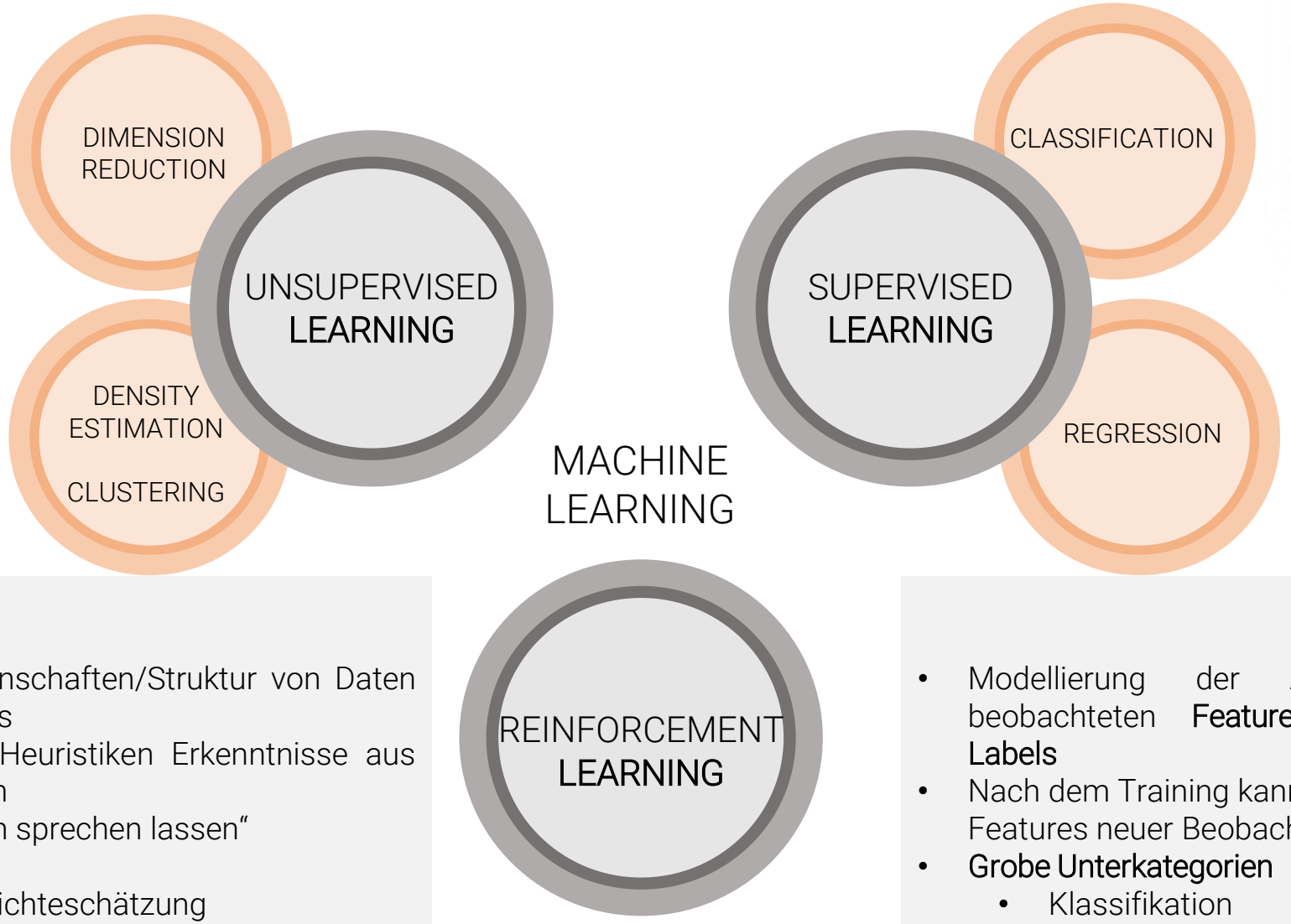
So what?

Wir führen in dieser Vorlesung die wichtigsten Konzepte des Machine Learning anhand einfacher und qualitativer Beispiele ein

Was denken Sie?

Was suchen also Machine Learning Modelle?

Kategorien des Machine Learning: Überblick



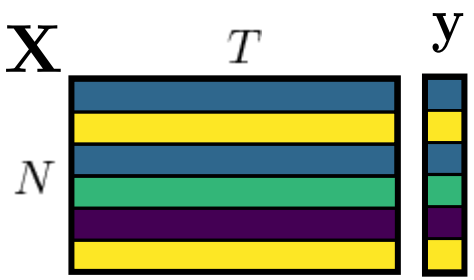
Unsupervised Learning

- Modellierung der Eigenschaften/Struktur von Daten **ohne** zugehörige Labels
- Man versucht durch Heuristiken Erkenntnisse aus den Daten zu gewinnen
- „Den Datensatz für sich sprechen lassen“
- **Grobe Unterkategorien**
  - Clustering und Dichteschätzung
  - Dimensionsreduktion

Supervised Learning

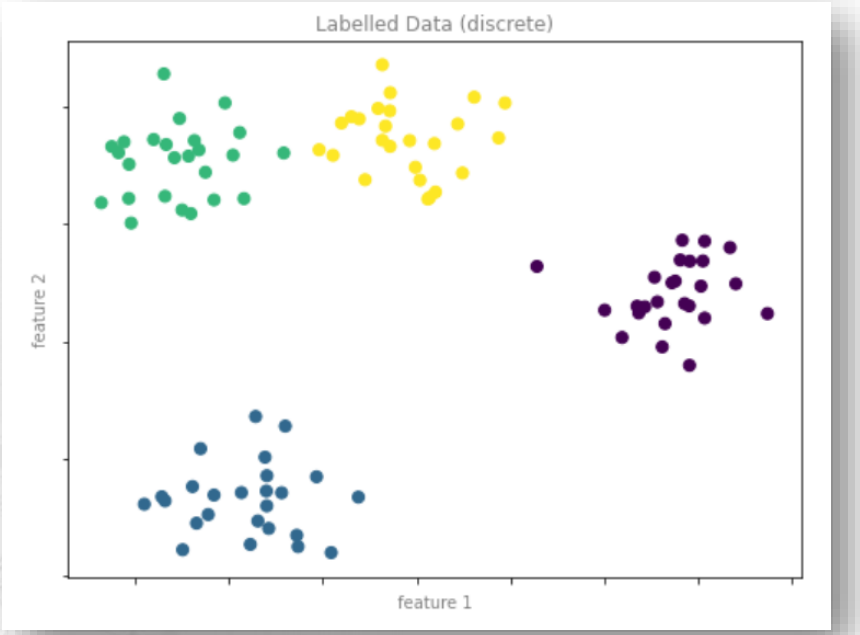
- Modellierung der Abhängigkeiten von beobachteten **Features** und zugehörigen **Labels**
- Nach dem Training kann das Modell Labels zu Features neuer Beobachtungen zuordnen
- **Grobe Unterkategorien**
  - Klassifikation
  - Regression

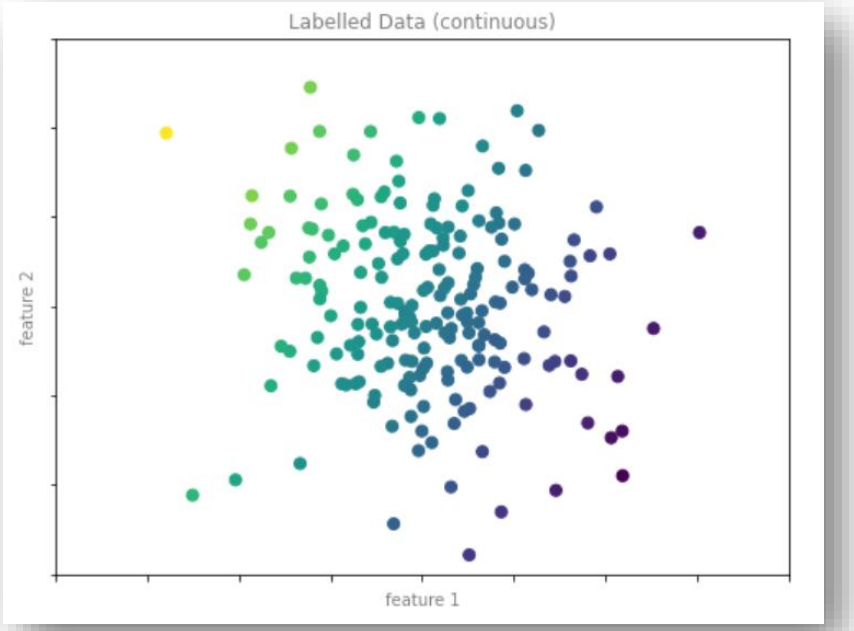
Supervised Learning: Datensicht



Labels bzw. Targets

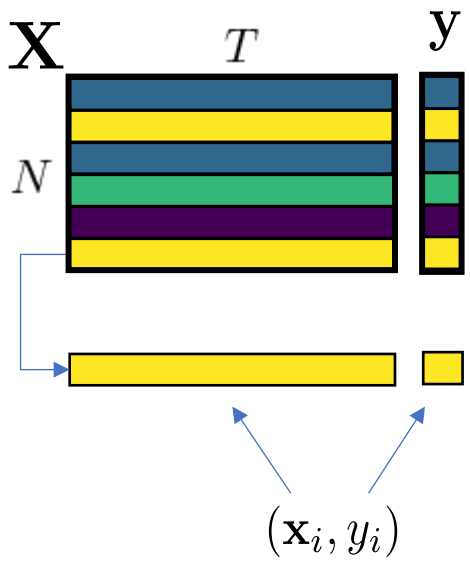
- Wir kennen schon unsere Feature-Matrix: nun assoziieren wir einen neuen Begriff mit dieser – ein **Label** oder **Target**
- Bei Labels handelt es sich um eine Variable, die **abhängige Variable**, die man aus den Features, den **unabhängigen Variablen**, vorhersagen will
- Labels können unterschiedliche Ausprägungen haben
  - **Klassifikation**: Labels haben eine **diskrete** Ausprägung
  - **Regression**: Labels haben eine **kontinuierliche** Ausprägung



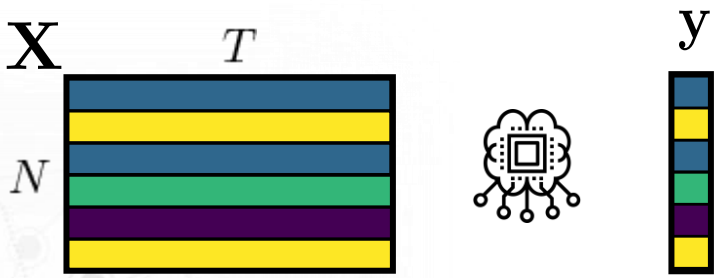


In den Abbildungen:
Farbe == Label

Supervised Learning: mathematische Notation



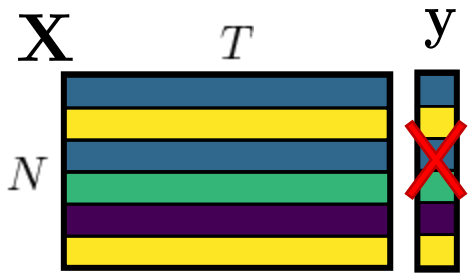
- Unsere Feature-Matrix beschreiben wir – wie gewohnt – durch ein $\mathbf{X}$ mit den Dimensionen *N x T*
- Das zugehörige Label ist ein *N*-dimensionaler Spaltenvektor $\mathbf{y}$
- Im sog. **Trainingsdatensatz** existiert zu jeder Beobachtung (Zeile der Feature-Matrix) ein Eintrag im Label-Vektor
- Wenn wir die Feature-Matrix als eine Menge an gestapelten Zeilenvektoren $\mathbf{x}_i$ betrachten, dann liegt also folgende Assoziation vor $(\mathbf{x}_i, y_i)$

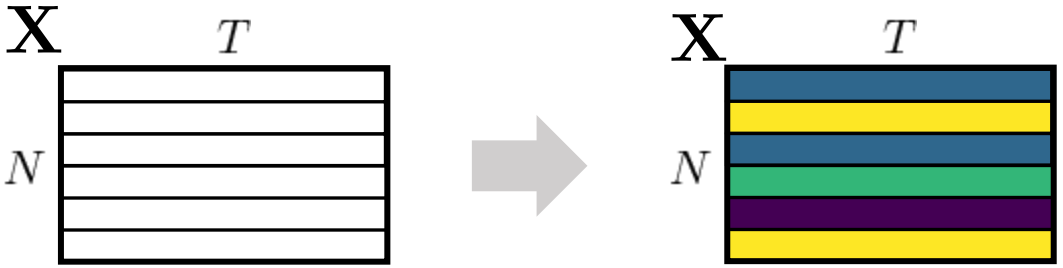


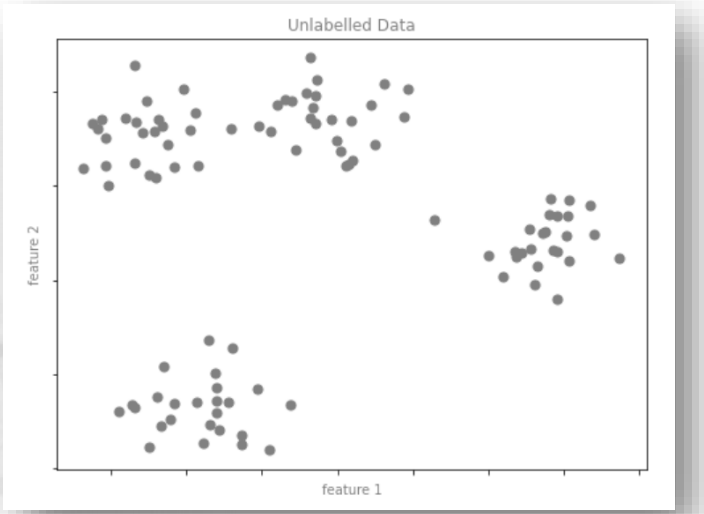
Trainingsdatensatz

- Unter diesem Begriff versteht man die Daten (und Labels), anhand derer das Machine Learning Modell **trainiert** wird
- An diesem wird also die Abhängigkeit der Labels von den Features geschätzt

Unsupervised Learning: Datensicht



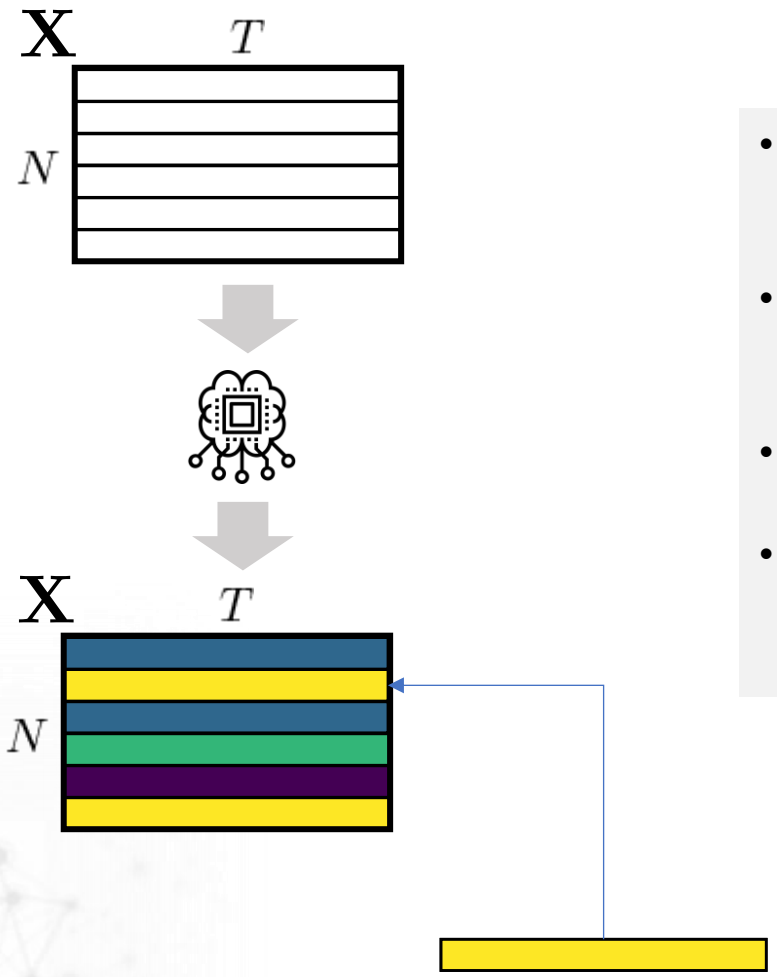




Unsupervised Learning == Keine Labels!

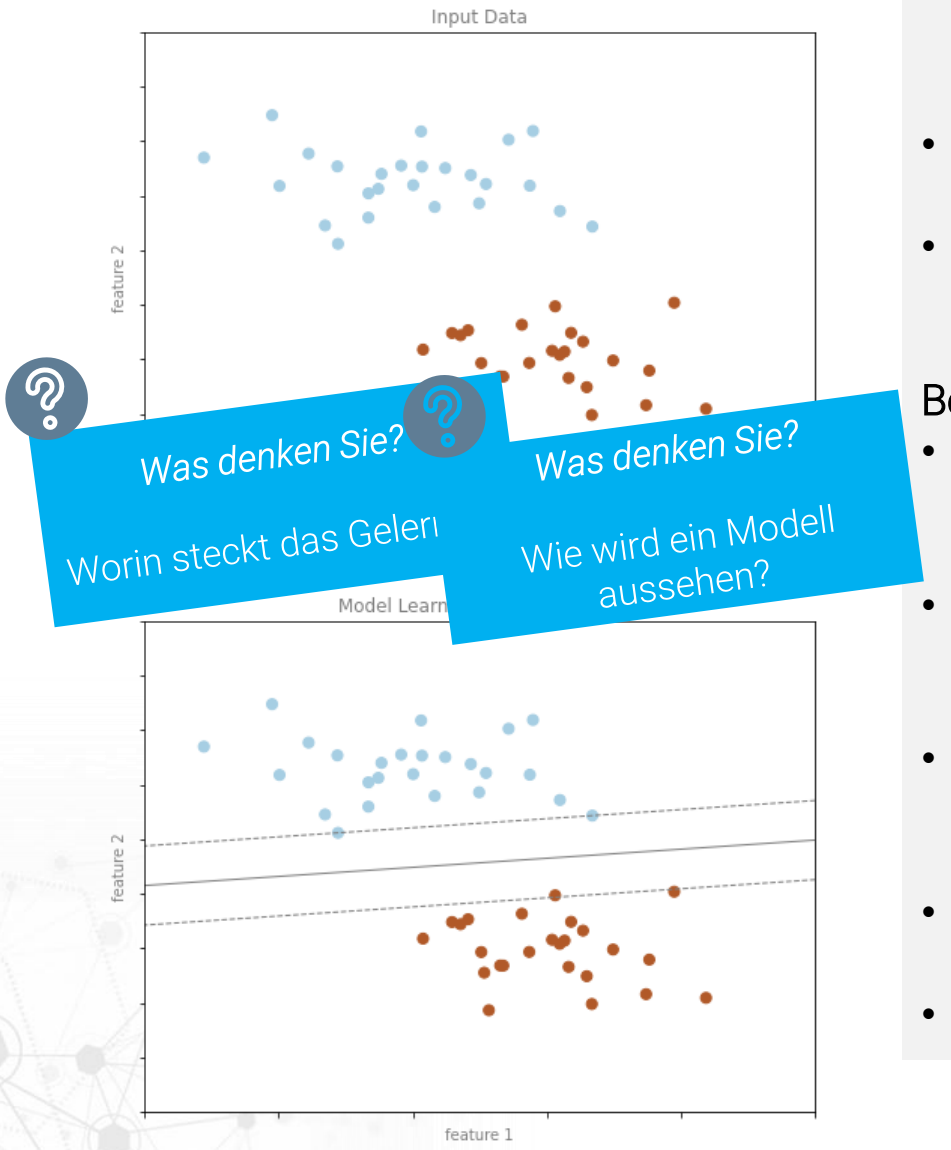
- Beim Unsupervised Learning liegen uns keine Labels vor
- Die, dem Datensatz zugrundeliegende Struktur, ist für uns (anfänglich) nicht ersichtlich bzw. markiert
- Wir haben es also mit einer „nicht eingefärbten“ Feature-Matrix zu tun
- Durch unsere Unsupervised Learning Modelle versuchen wir die Einfärbung zu schätzen

Unsupervised Learning: mathematische Notation



- Wir haben also im Unsupervised Learning Fall (zuerst) nur eine Feature-Matrix $\mathbf{X}$
- Das Machine Learning Modell versucht in diesem Fall Strukturen bzw. Muster in den Daten zu entdecken
- Wir können diese Muster dann auch mit Labels $\mathbf{y}$ versehen
- Nachdem das Modell trainiert wurde, können auch im Unsupervised Learning Fall neue, ungesehene Daten den entdeckten Strukturen bzw. Muster zugeordnet werden

Supervised Learning: Klassifikation



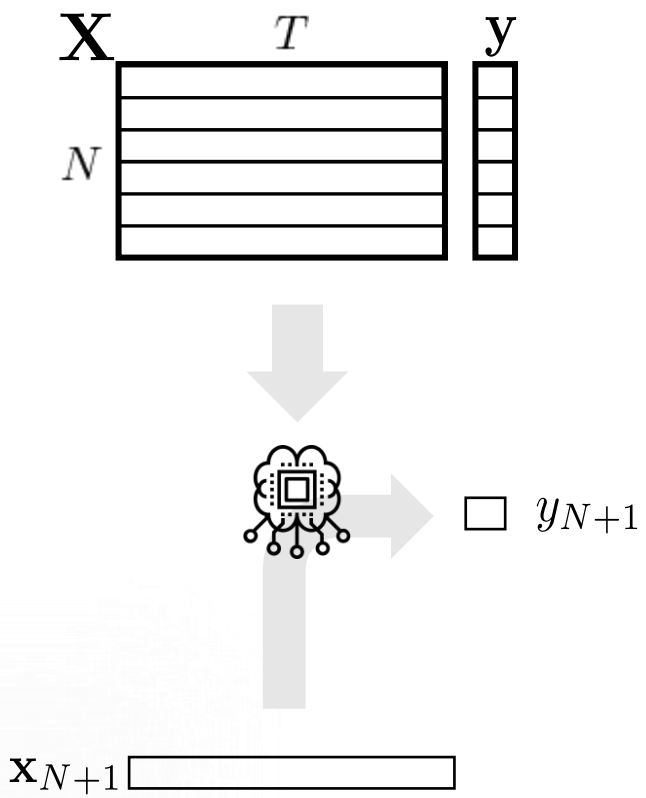
- Unsere Aufgabe bei einer Klassifikation ist: wir haben einen gelabelten Datensatz vorliegen (Trainingsdatensatz)
- Daran trainieren wir unser Machine Learning Modell
- Unser Modell soll dann **ungelabelte** Datenpunkte **klassifizieren** → Labels zuweisen

Beispiel

- In unserem Beispiel liegen zweidimensionale Daten vor, die auf zwei Kategorien aufgeteilt sind (blau und rot)
- Unser Modell soll soz. eine „Trennlinie“ finden, die die beiden Klassen aufteilt
- Neue Daten können dann anhand dieser Trennlinie bewertet werden, ob sie in die blaue oder rote Kategorie fallen
- Die **Modellparameter** wären in diesem Fall die Koeffizienten der Geraden
- Diese Modellparameter werden aus den Daten **gelernt**

So what?
„Learning is finding parameters.“

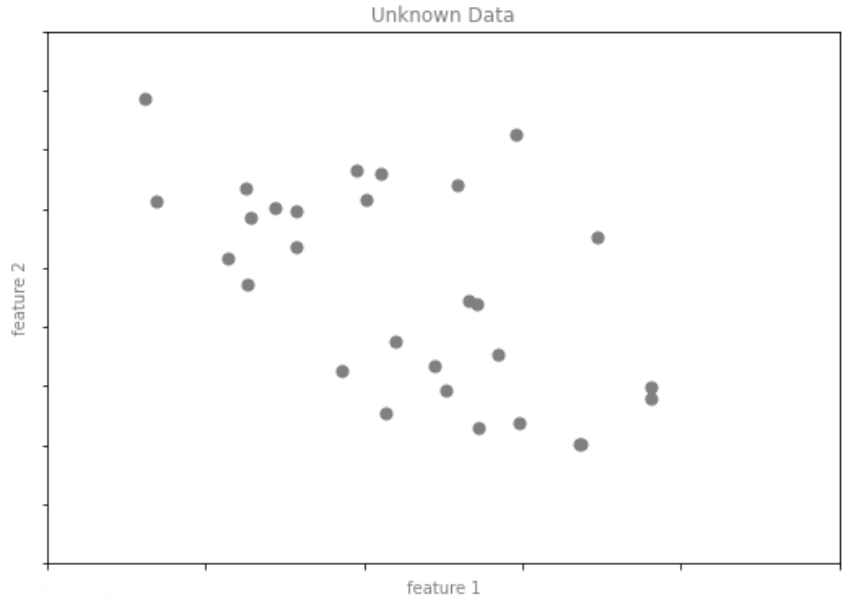
Konzepte des Machine Learning: Scoring bzw. Prediction

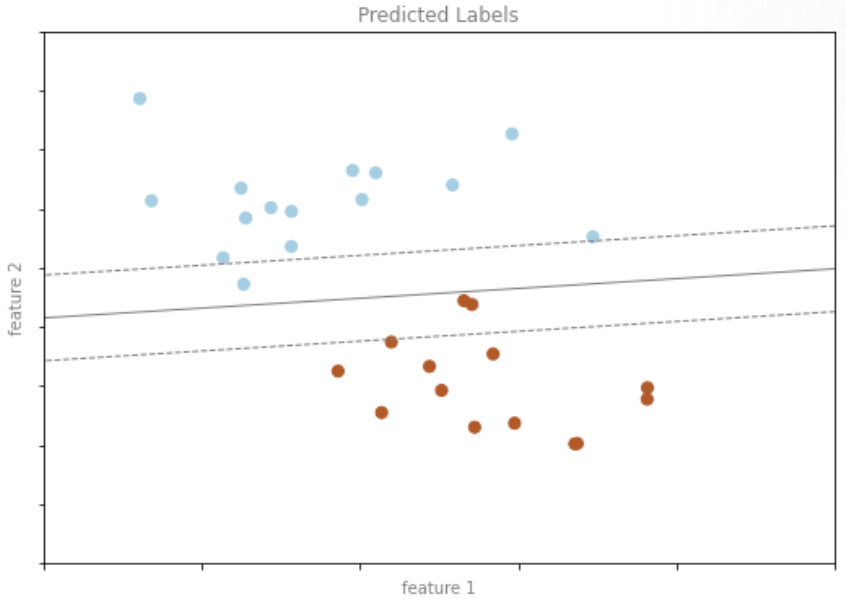


- Wir haben nun schon kennengelernt, dass aus dem Training ein Modell entsteht, das die Beziehung zwischen Features und Labels abbildet
- Ziel beim Machine Learning ist dieses gelernte Wissen auf **neue**, vorher **ungesehene** Daten, anzuwenden
- Man führt dem Modell neue Beobachtungen zu und dieses bewertet diese dann
  → Das Modell weist den neuen Beobachtungen Labels zu

Diese Zuweisung von Labels auf neue Beobachtungen nennt man Scoring bzw. Prediction

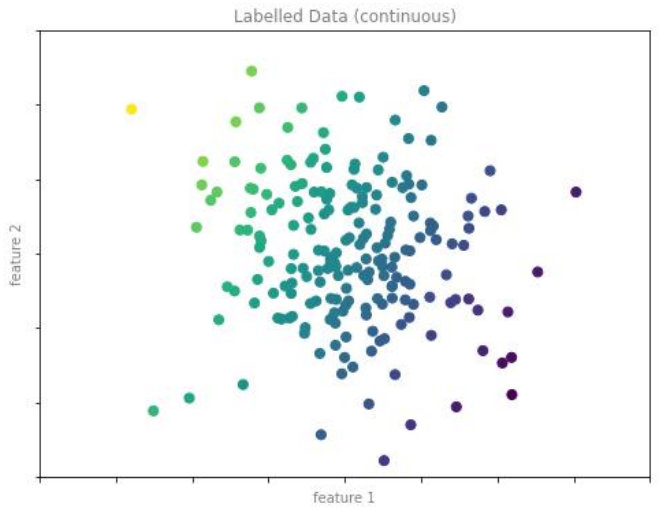
Supervised Learning: Klassifikation

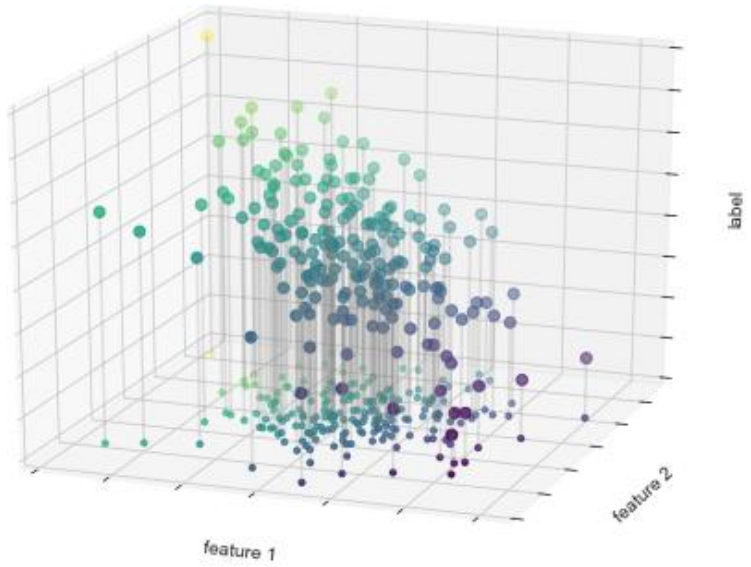




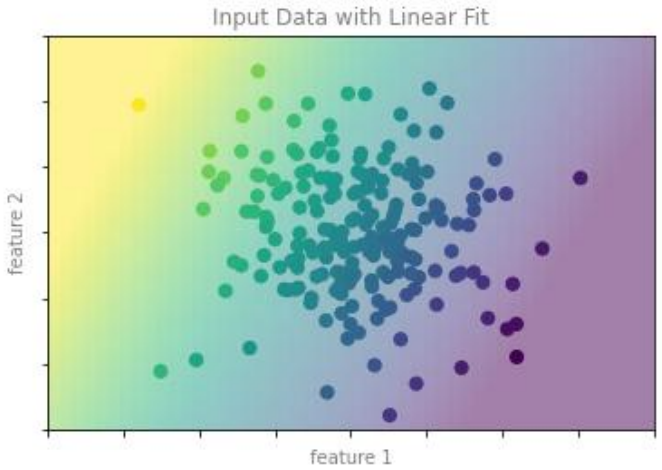
- Scoring bzw. Prediction wird in unserem Beispiel so durchgeführt, dass neue, unbekannte Daten mit dem Modell (unsere gelernte Gerade) **verglichen** werden
- Je nachdem auf welcher Seite der Gerade die Datenpunkte liegen, werden die entsprechenden **diskreten** Labels (rot oder blau) zugeordnet

Supervised Learning: Regression

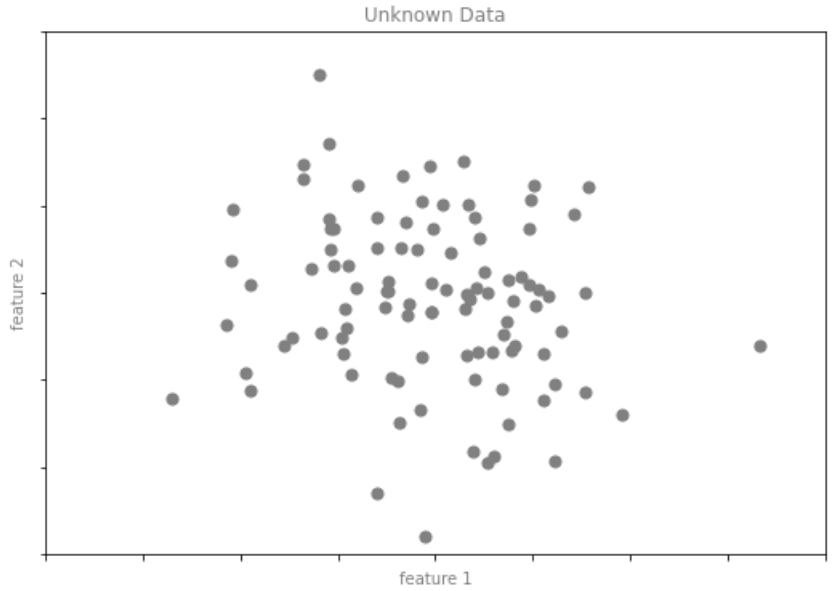


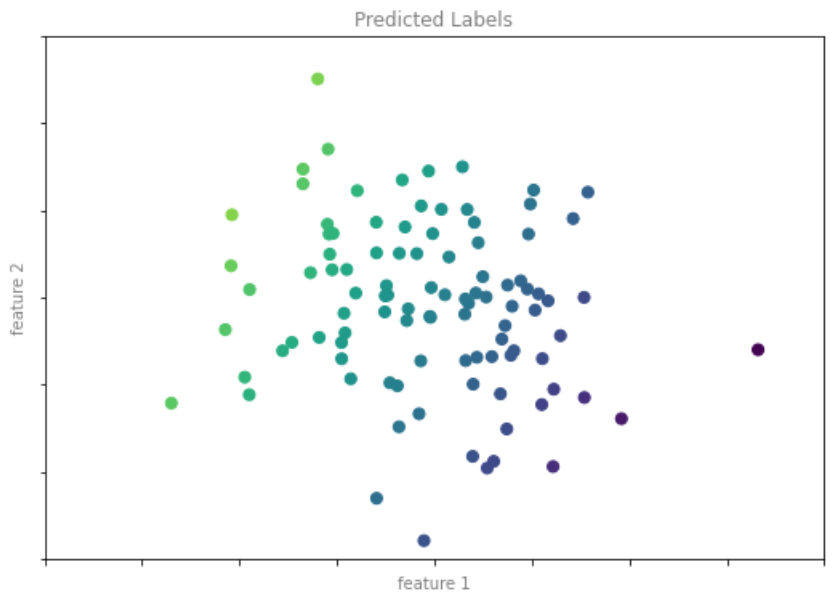


- Im Gegensatz zur Klassifikation liegen uns bei der Regression **kontinuierliche** Labels vor
- In unserem zweidimensionalen Feature-Raum können wir uns das Label dann als eine **dritte Dimension** vorstellen
- Bei einer einfachen Regression würde man dann versuchen eine **Ebene** auf diese Daten zu fitten



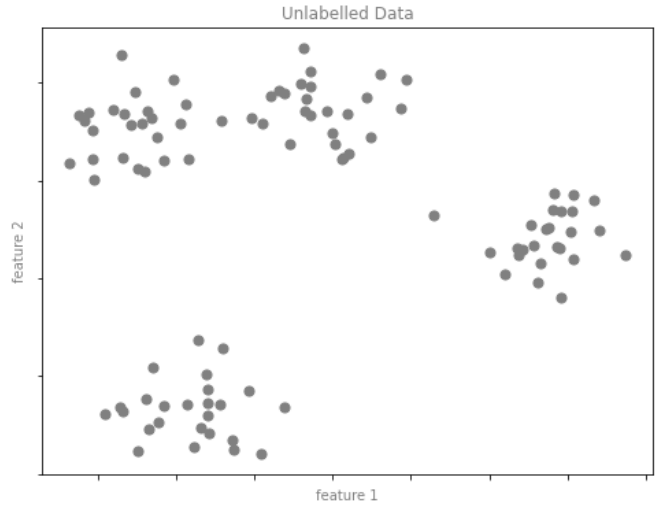
Supervised Learning: Regression: Scoring

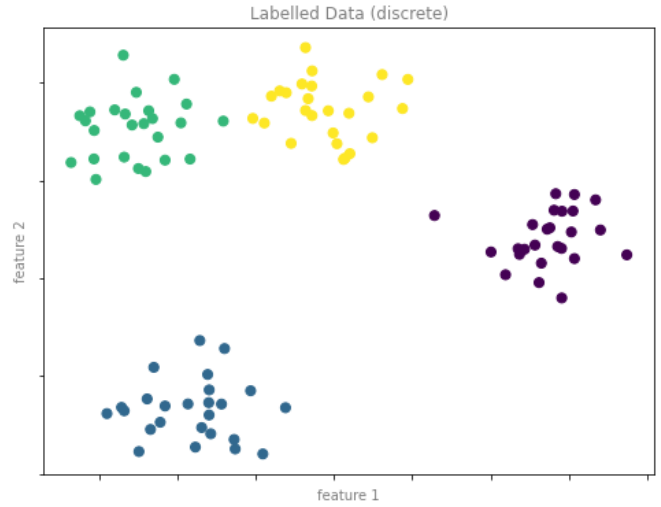




- Beim Scoring im Regressionsfall werden die unbekannten Daten soz. auf die geschätzte Ebene **projiziert**
- Dies liefert uns dann die **neuen**, kontinuierlichen **Labels** für die ungelabelten, neuen Daten

Unsupervised Learning: Clustering





Was denken Sie?

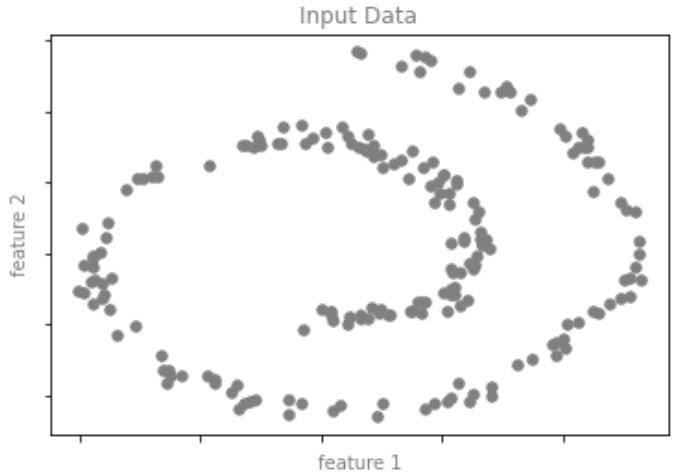
Wie würden Sie abschätzen, ob ein Datenpunkt zu einem bestimmten Cluster gehört? Welches Maß würden Sie verwenden?

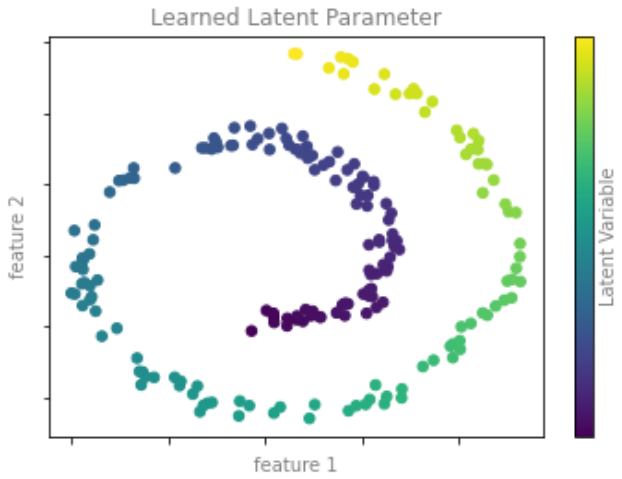
- Ein typisches Unsupervised Learning Problem ist **Clustering**
- Beim Clustering werden die ungelabelten Daten **diskreten Gruppen bzw. Labels** zugeordnet
- Clustering ist eng verwandt mit der **Dichteschätzung** der zugrundeliegenden Daten
- Neue Datenpunkte können dann z.B. durch **Abstandsbetrachtungen** einem bestimmten Cluster zugewiesen werden

Was denken Sie?

Welches Problem sehen Sie hier?

Unsupervised Learning: Dimensionsreduktion





- Ein weiteres Problem des Unsupervised Learnings ist die Dimensionsreduktion
- Ziel ist eine **niedrigdimensionale** Repräsentation der Daten, die die meisten/wichtigsten Eigenschaften dieser erhält
- In unserem Beispiel sehen wir schon visuell, dass eine bestimmte Struktur in den Daten vorhanden ist
- Diese Daten liegen auf einer Spirale – sind also intrinsisch **eindimensional**!
- Die Farbe entspricht einer sog. **latenten Variable** → Eine neue Koordinatenachse, die das Modell entdeckt/gelernt hat

Machine Learning in Python

`scikit-Learn`