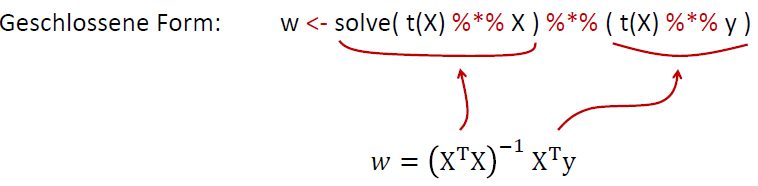
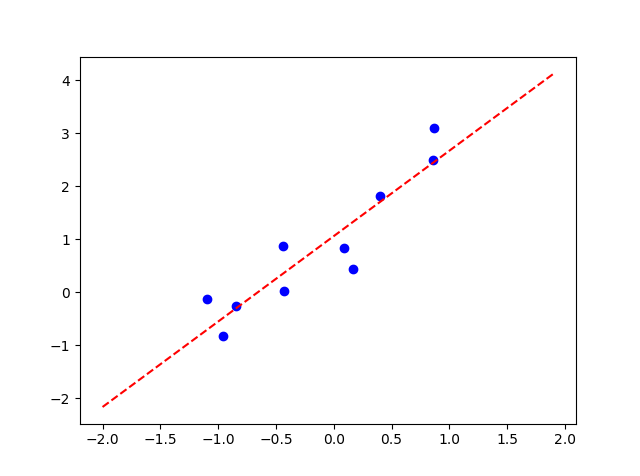
**1.Lineare Regression**

**1.1Training mittels abgeschlossener Lösung**

**Formeln:**



**Graph:**

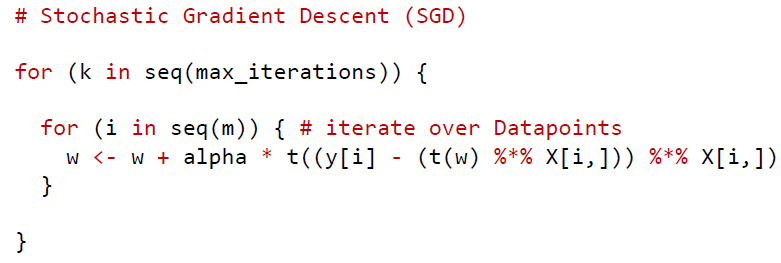


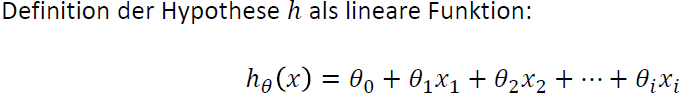
**Frage:** Was ist der naheliegende Nachteil an der abgeschlossenen Lösung, sobald ein

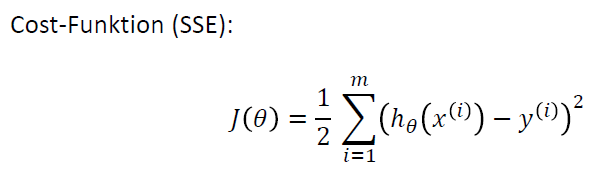
besonders großer Datensatz verwendet werden soll?

**Antwort:**

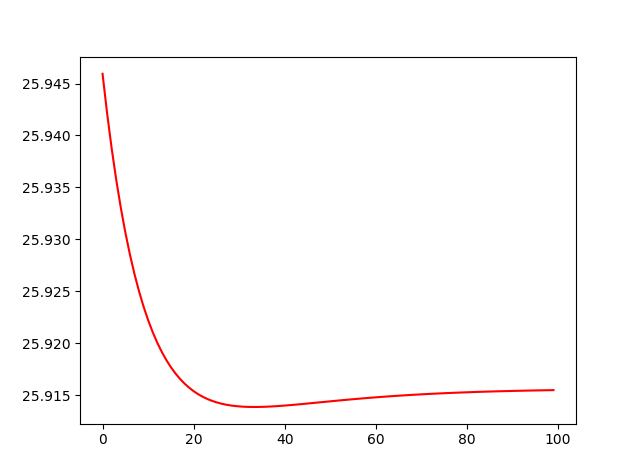
**1.2 Training mittels Stochastic Gradient Descent (SGD)**

**Formeln:** 





**Graph:**



Probieren sie unterschiedliche Werte für die Parameter alpha und iterations in der predict()-Funktion aus und betrachten Sie die Auswirkung auf den Graphen des Cost-Wertes. Dokumentieren Sie knapp Ihre Beobachtungen.

**Frage:** Warum wird in der Praxis Stochastic Gradient Descent (SGD) gegenüber Batch

Gradient Descent (s. Vorlesung) bevorzugt?

**Antwort:** BGD geht über das gesamte Trainingsset um Parameter 𝜃𝑗 zu aktualisieren

* Langsam, wenn Datensatz RAM sprengt

SGD geht über das Trainingsset und verändert die Parameter gemäß der Gradienten des betrachteten Datensatzes i

* Stochastisch durch Annäherung an Subset der Daten
* Optimiert durch mini-batches
* Übertritt lokales Optimum durch „noisy“ Gradient

**1.3 Vorhersage von Immobilienpreisen mit linearer Regression**

Dokumentieren Sie kurz, welche Implementierung der linearen Regression Sie verwendet haben und

nehmen Sie einen passenden Plot in das Protokoll auf.

linear\_regression\_sgd

**Formeln:**

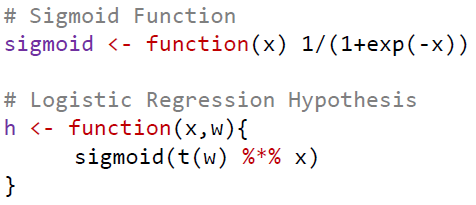
**Graph:**

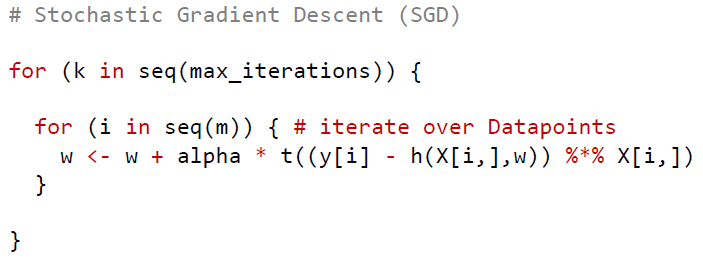
**2. Logistische Regression**

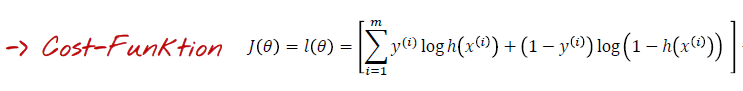
**2.1 Training mittels SGD**

Überprüfen Sie, welche Auswirkung die Veränderung des Wertes iterations und alpha in der Funktion fit() aus die Qualität der Decision-Boundary haben und dokumentieren Sie ihre Ergebnisse im Protokoll.

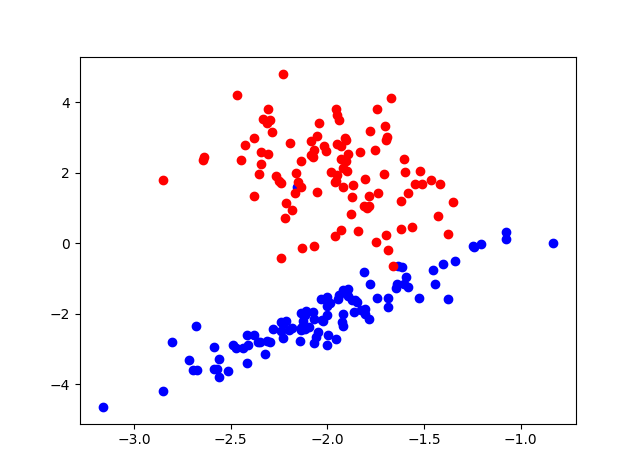
**Formeln:**







**Graphen:**

****

**Frage:** Was müssten Sie tun, damit Sie mehr als zwei Klassen vorhersagen können?