In der vorherigen Aufgabe wurde die Erfolgsmessung durch Ausführungen und Vergleich der Ergebnisse manuell durchgeführt. In dieser Aufgabe soll dieses Vorgehen automatisiert werden.

**Ziele**

* Implementierung von ML basierten SW Systemen
* Betrieb und Ablauf von ML basierter Software transparent gestalten
* Automatisches Testen von ML basierten SW Systemen

**Teilaufgaben**

Lesen und verstehen Sie den folgenden Artikel über das Logging und einen einfachen Input – Output unit test Ansatz ML basierter SW Systeme.

<https://towardsdatascience.com/unit-testing-and-logging-for-data-science-d7fb8fd5d217>

Ihre Aufgabe ist, den Ansatz auf ein **Übungsbeispiel** / **Projekt** des obigen Python Kurses zu übertragen. Gehen Sie wie folgt vor:

Wählen Sie ein Übungsbeispiel aus dem obigen Python-Kurs, z.B. das DeepLearning Übungsbeispiel mit dem MNIST Datensatz. In der vorherigen Aufgabe haben Sie ggf. dieses Beispiel auf myBinder erfolgreich ausführen können.

Legen Sie ein neues Git Repository für diese Aufgabe an. Prüfen Sie, dass Sie das Beispiel via myBinder erfolgreich ausführen können.

Implementieren Sie nun die **my\_logger** und **my\_timer** Funktionen aus dem obigen Artikel und wenden Sie sie auf von mindestens eine von Ihnen gewählte Funktion des Beispiels an.

Dokumentieren Sie Ihr Ergebnis in der README.md Datei.

Mittels der Funktionen sollen nun zwei Testfälle implementiert werden; einmal für die Vorhersagefunktion des Modells, **predict(),** und einmal für die Trainingsfunktion des Modells, **fit().** Schreiben Sie folgende zwei Testfälle.

**1. Testfall:** Wählen Sie geeignete Indikatoren, die Ihnen anzeigen, dass die Vorhersagefunktion predict() des Modells korrekt funktioniert. Im Artikel nutzt der Autor die Indikatoren **Accuracy** und **Confusion Matrix**. Schreiben Sie für **predict()** einen Testfall auf ausgewählten Testdaten. Die Testdaten legen Sie in einem **Testdatenfile** ab.

**2. Testfall:** Überprüfen Sie, dass das System innerhalb normaler Parameter läuft, indem Sie die Laufzeit der **Trainingsfunktion fit()** testen. Loggen Sie dazu eine repräsentative Laufzeit mit dem obigen **Wrapper**. In dem Testfall überprüfen Sie, dass die Laufzeit der Trainingsfunktion während der Testfallausführung einen Grenzwert, z.B. **120%** der repräsentativen Laufzeit, **nicht überschreitet**.

Lassen Sie beide Testfälle ausführen und dokumentieren Sie die Bildschirmausgabe in der README.md. Dokumentieren Sie wie ein Nutzer die Testfälle mit dem Testdatenfile ausführen kann.

* Dokumentation der Bildschirmausgabe wie oben beschrieben
* Der Prüfer des Kurses kann beide Testfälle m.H. Ihrer Dokumentation und dem Testdatenfile erfolgreich über mybinder ausführen.